Transactions of the I·R·E

Professional Group on ELECTRONIC COMPUTERS

June, 1954

Volume EC-3 Number 2

TABLE OF CONTENTS

The Institute of Radio Engineers

IRE PROFESSIONAL GROUP ON ELECTRONIC COMPUTERS

The Professional Group on Electronic Computers is an association of IRE members with professional interest in the field of Electronic Computers. All IRE members are eligible for membership, and will receive all Group publications upon payment of the prescribed assessment.

1954 Assessment: $2.00

Published by the Institute of Radio Engineers, Inc., for the Professional Group on Electronic Computers at 1 East 79th Street, New York 21, N. Y. Responsibility for the contents rests upon the authors and not upon the Institute, the Group, or its Members. Extra copies of this issue are available for sale to IRE-PGEC members at $1.65; to other IRE members at $2.45; and to nonmembers at $4.95. Address requests to The Institute of Radio Engineers, 1 East 79th Street, New York 21, N. Y.

Notice to Authors: Address all papers and editorial correspondence to R. E. Meagher, 168 Engineering Research Laboratory, University of Illinois, Urbana, Ill. To avoid delay, 3 copies of papers and figures should be submitted, together with the originals of the figures which will be returned on request. All material will be returned if a paper is not accepted.

PGEC NEWS

Edited by

Stanley B. Disson
Burroughs Corporation, Research Center
Paoli, Penna.

During the past year the membership of the PGEC increased by over 1000, with the total paid member- now close to 2500. Active chapters have been ablished in Los Angeles, San Francisco, Philadelphia, hington, D.C., New York, Chicago and Albuquerque. pters have been approved for Boston and Detroit, and rts are being made to organize chapters in other sec- s of the country.

The Eastern Joint Computer Conference, which was l in Washington, D.C., in December, and the Western nt Computer Conference, which was held in Los eles in February, each had registrations of approxi- ely 1000. C. W. Adams of M I T has been named irman of the 1954 Eastern Joint Computer Conference ch is scheduled for Philadelphia in December. The irman of the next Western Joint Computer Conference Villiam Martin of Telecomputing Corporation.

PGEC ADMINISTRATIVE COMMITTEE ELECTIONS

The following officers and new members of the EC Administrative Committee were elected at the ual Administrative Committee meeting on March 25, e effective July 1, 1954.

hairman:	Harry T. Larson
ice-Chairman:	James R. Weiner
ew Members:	Isaac L. Auerbach
	Werner Buchholz
	Bernard M. Gordon
	William L. Martin
	Jerre D. Noe.

SECTIONAL ACTIVITIES

Chicago. The activities of this chapter were initiated eptember 1953 with a talk by R. J. Kleen, Argonne oratories, on "Electrostatic Memory Tubes." J. E. ertson, University of Illinois, spoke on "Logical ign of Digital Computers" in November and L. D. nson, IBM, discussed the IBM Type 650 Magnetic n Calculator in December. In February, David Ruben- , Armour Research Foundation, discussed "The ation of a Computing Center" and A. J. Fitzpatrick, roughs Corp., spoke on "The Multi-Output Beam tching Tube." The April meeting was devoted to og computers.

Albuquerque. As with most of the other chapters, monthly meetings are held except during the summer months. This chapter held its organizational meeting in January 1954 and the papers presented through May are: "The Analog Computer as an Automatic Computing Machine," by F. Lane; "The Design of an Analog Computer," by R. McGehee; "General Purpose Digital Computers," K. Ball; and "Engineering Applications of Boolean Algebra," by J. E. Gross. All of the speakers are with the Sandia Corporation.

Philadelphia. Of particular interest is the experimental meeting held in January at which a panel discussed "Design Criteria for Digital Computer Circuits." The panel included N. H. Taylor of MIT, R. E. Meagher of the University of Illinois, and J. L. Hill of Engineering Research Associates. Three prominent local engineers then presented a prepared discussion of the papers prior to a general question period. In cooperation with the local sections of the IRE and AIEE, this chapter also sponsored the Digital Storage Devices Symposium during February and March. Six papers were presented on the more important storage devices.

Los Angeles. In December D. H. Raudendush, R. H. Lawlee, E. R. Quady, and T. C. Alrich described the Consolidated Engineering 36-101 Computer, and in January, H. D. Huskey of the National Bureau of Standards spoke on "New Topics in Coding and Programming."

New York. One of the new chapters, this group encompasses the New York City and Long Island areas and also includes the Connecticut Valley members on its mailing list. In January, C. Andrews and E. L. Schmidt of Teleregister Corporation, and C. Ammond of American Airlines discussed and demonstrated the Reservisor installation at LaGuardia Airport. In March, W. B. Groth of Bell Telephone Laboratories discussed "Automatic Message Accounting." W. A. Malthaner, H. F. Vaughan, J. R. Anderson and Q. W. Simkins discussed and demonstrated Ferroelectrics and Transistor Computer Circuits in April, and the Univac System was the topic for May. C. W. Wockenfuss of Control Instrument Company will speak on "High Speed Printing" on June 29.

Washington, D. C. In January C. E. Miller of Engineering Research Associates described the 1103 computer. The February meeting was devoted to a discussion on "A Temperature-Invariant Solid, Ultrasonic Delay Line" by J. Filler of the Diamond Ordnance Fuse Laboratory. "The IBM Type 650 Magnetic Drum Calculator" by E. Hughes of IBM was the topic for March.

LOGIC, DISCOVERY, AND THE FOUNDATIONS OF COMPUTING MACHINERY

M. E. Maron
International Business Machines Corp.
San Jose, California

SUMMARY – This paper describes the logical nature of computing machines in terms of languages and the types of problems that can be solved by logical operations on languages. The problem of *discovery* in mathematics and empirical science is discussed, and an "inductive" machine is described which would be able to formulate hypotheses, modify them in the light of new experience and eventually discover the laws of a very simple universe.

1. INTRODUCTORY REMARKS

1.1 *Aim and purpose.* In recent months there has been an increasing interest, among those concerned with the logical foundations of computing machinery, in the possibility of constructing (or programming) a computer which will learn.[1] To speak of learning, without qualification, is too general; usually the problem is posed as follows: "Can one design a device which will learn a winning strategy for a game?" or "Can one build a machine that can learn the rules of a game?" The problem of learning the rules of a game corresponds, in a loose sense, to the problem of discovering physical laws in science, and we wish to determine whether a machine can be designed that will discover. In order to clear the ground for this inquiry, it is necessary to describe the logical operations that machines can perform, and also to examine what is meant by "discovery" in mathematics and empirical science. It is the purpose of this paper to present a clarified conception of the logical foundations and functions of modern computing devices. This perspective should enable us to see in what sense, if any, machines may be constructed to perform radically different operations, and also to see whether computers are fundamentally limited as to possible operations.

1.2 *Computers as language transducers.* What is a computing machine? In general, the term "computer" connotes a mechanism whose application is restricted to arithmetic or numerical operations, but we shall see that modern automatic machines are able to process nonnumerical as well as numerical information and are able to operate on information in ways other than mere arithmetic manipulation. In fact, it is because data handling machines are potentially capable of performing so many different operations with such a variety of information that they have been compared with brains. We shall have more to say about automatic information handling instruments and their analogy with brains, but in the meanwhile the term "computer" will be used synonymously with "automatic information handling device" to give it a broader meaning.

Automatic computers are, in essence, syntax machines.[2] The word "syntax" refers to the formal relationships existing between given expressions of a language, and to state that computers are syntax machines is to state that computers perform logical operations on expressions (sentences) of a language. It is only when knowledge (information) is formulated in terms of a language that it may be stored, analyzed, logically manipulated and communicated; hence, we must refer to languages and how machines manipulate linguistic expressions. If machines are to be understood as language transducers, we must state clearly what a language is and how it contains rules for the manipulation of linguistic expressions.

2. INFORMATION MACHINES AND LANGUAGE

2.1 *Description of a formal language.* In order to describe a language, it is necessary to enumerate those symbols with which expressions in the language are constructed. These building blocks of language are called "primitive symbols," and it is in terms of these that other symbols (defined symbols) are introduced into the language. Just as there are primitive and defined symbols in a language, so also there are primitive and defined operations. Given the class of symbols and operations, we can now specify what combinations (of symbols and operations) constitute meaningful expressions in the language. The rules which specify the class of meaningful (well-formed) expressions are called "formation rules." For example, if the primitive and defined symbols of our language are the integers from 1 to 8 inclusive and the operations are those of addition, subtraction and the relation of equality, then the formation rules specify that such expressions as "1 + 3 = 4" and "1 - 6 = 3" are wellformed expressions, whereas " 1 = " and " - 3 + " are not well-formed expressions.[3] (Although natural languages, such as English, are not strict and hence do not have explicit formation rules, we might say that the rule

[1] W. Ross Ashby, "Can a mechanical chess-player outplay its designer?" *Brit. Jour. Phil. Sci.*, vol. 3, no. 9, pp. 44-57; 1952.
---------- "Design for a Brain," John Wiley and Son; 1952.
A.G. Oettinger, 'Programming a computer to learn," *Phil. Mag.*, vol. 43, pp. 1243-1262.
P. I. Richard, "On game-learning machines," *Sci. Monthly* vol. 74, no. 4, pp. 201-205; 1952.
C. E. Shannon, "Programming a computer for playing chess," *Phil. Mag.*, vol. 41, pp. 256-275; 1950.
A. M. Turing, "Computing machines and intelligence," *Mind*, vol. 59, no. 236, pp. 433-460; 1950.

[2] G. W. Patterson, "Logical syntax and transformation rules," *Proc. Second Symposium on Larger Scale Digital Calculating Machines*, Harvard Univ. Press, p. 125; 1951.

[3] Although the expression "1 - 6 = 3" is false, it is *meaningful*. See H. Reichenbach, "Experience and Prediction," Univ Chicago Press, ch. 1, 1933.

grammar which states that every meaningful sentence st have a subject and a predicate corresponds to what mean by a formation rule.)

Our description of a formal language is not complete ess we describe those rules that state which logical nipulations may be made on expressions of the langu-e. Such rules are called "transformation rules." For ample, if we are given the expressions "8/2 = 4" and + 1 + 1 + 1 = 4", we may substitute "equals for uals" and obtain the new expressions "8/2 = 1 + 1 + + 1". The rule which states that one may substitute uals for equals is what we call a "transformation rule."

In deductive logic the use of the transformation rules generate new expressions from given expressions as-res us that the derived propositions are true, provided initial ones are true. That is to say, the transforma-n rules guarantee that the property of truth will be pre-rved in making logical transformations. (Later we shall k whether there are transformation rules of *inductive* gic which guarantee that highly *probable* conclusions derivable from true information.)

It is necessary to clarify the notion of transforma-n (or inference) rules which are *descriptions* of the owable logical manipulations rather than *prescriptions* escribing in what sequence the allowable operations are be made in order to obtain the desired conclusion. us given our input information (postulates), the trans-mation rules do not tell us *how* to proceed in order to tain a certain output (conclusion); they tell us only at certain logical transformations are allowable and ners not.

The sequence of transformations from the postulates a theorem is called a proof or a derivation if, at each ep in the sequence, there is a justification of that ep by reference to one of the specified inference rules.

2.2 *Computers and language transformations.* Refer-g again to the input as the postulates and the output the derived conclusion, the coding problem is that of nslating the information from a given language to well-med expressions in the machine language. The import-t aspect in the operation of a data processing machine that of programming. The program is a list of the trans-mation rules in the order in which they are to operate the given information. This sequence is analogous to e sequence of steps in the proof of a mathematical eorem. Thus, the programmer must instruct the machine what sequence it is to follow the transformation rules order to arrive at a desired conclusion. This point is ry important, and we shall refer to it again.

Since data handling instruments are language ma-ines and operate on information *according to a given rogrammed) sequence of orders,* (i.e., they obey rules), nat *types* of problems can they solve? We know that all oblems of *computation* involve strict rules. There are les which tell one how to find the sum of two numbers, the value of a number defined by a definite integral to decimal places. What other problems require strict rules for their solution? For what class of problems is it impossible to find rules? Does *learning* consist of altering rules of operation on the basis of new information? Let us turn to consideration of these questions.

3. TYPES OF PROBLEMS

3.1 The notion of an effective procedure. The class of all possible problems can be divided into two exclusive categories which are as follows:

1. Those problems for which there is an effective procedure of solution.
2. Those problems for which there is *not* an effective procedure of solution.

Carnap defines an "effective procedure" as follows: "A Procedure is called 'effective' if it is based on rules which determine uniquely each step of the procedure and if in every case of application the procedure leads to the solution in a finite number of steps."[4] We have seen that a problem can be solved by a conventional computer if there is an effective procedure of solution. That is to say, if we want a *guarantee* that the computer will find the solution in a finite time, we must show that the programmed problem has an effective procedure of solution. If this is the case, let us turn to the key problems in deductive and inductive logic and determine whether they have an effective decision procedure, and then see how it is possible that some types of problems are solvable without having an effective method of procedure.

3.2 *Deductive and inductive logic.* In deductive logic, we reason from some given true information to other true information by the use of certain transformation rules. The fundamental concept of deductive logic is that of *implication,* and we say that an argument is *valid* if, and only if, the premises (input information) *imply* the conclusion (output information). To say that A implies B is to say that it is impossible for A to be true and B false. The process of deduction can never supply new information, i.e., if an argument is valid and hence the conclusion follows from the premises, then the conclusion was contained in the premises in a logical sense, and it does not supply us with any logically new information. (Of course, the conclusions are not always *obviously* contained in the premises and, therefore, the conclusion is often new in a psychological sense.)

The nature of inductive logic is quite different, and there is a current controversy concerning the logical foundations of inductive reasoning. However, we may say that in inductive logic the fundamental concept is *probability* and we start with information which we know or assume to be true and then obtain other *new* information which is *probable,* relative to the given information. For example, in science we formulate an hypothesis to account for some observed data, and if the hypothesis is good we can deduce not only the given data but other data

[4] R. Carnap, "Logical Foundations of Probability," Univ. Chicago Press; 1950.

as well (predictions). *Hence, induction provides new information.* Thus, we extend our knowledge by induction and not by deduction, but whereas we can be certain of the truth of a conclusion which is correctly deduced from true information, we lose this guarantee of truth in inductive reasoning and we find that an inductive conclusion is only probable.[5] Let us now turn to the *three* fundamental problems in both fields of logic.

3.3 *First Problem: To find a conclusion.*[6]

3.3.1 *Deductive logic.* Assume that a mathematician is given a set of axioms for some geometry and is asked to derive a theorem from them which refers to a certain type of spatial configuration. How does he proceed? He does *not* proceed by following a set of rules, as he would if he were asked to compute the square root of some number. The point is that there is no effective method of solution for problems of this type. In order to find a conclusion from given axioms, the mathematician proceeds by certain vague, intuitive ways, not by fixed rules. The essential point is that the process of *discovery* in mathematics is not rational in the sense that it does not involve rules of logic, rather, it involves intuition and creative ability (which are not clearly understood by contemporary psychologists).

3.3.2 *Inductive logic.* Given some experimental data, how is an hypothesis reached which will explain factual data? For example, given some data which a physicist might obtain, how does he formulate a theory which will explain the observed phenomena? There is no set of fixed rules which guarantee that by following them one can arrive at the best, or even a good, hypothesis. As Einstein has stated: "There is no logical ways leading to these . . . laws, but only the intuition based on a sympathetic understanding of experience."[7] Most philosophers and scientists agree with Einstein that there is no effective procedure for finding fruitful hypotheses. Just as in the case of discovery of theorems in mathematics and logic, so also the discovery of hypotheses in empirical science is a matter of intuition and creative ability, and not a matter of obeying a set of fixed rules.

We see that if there *were* an effective method for discovering a conclusion in mathematics or for discovering hypotheses in physics, then there would be no unsolved problems in these areas. One would merely take the initial information and follow the rules of discovery and in a finite number of steps reach the desired conclusion. However, there is no effective procedure for the solution of this first class of problem, hence we cannot program a computer and be sure that the machine will present us with the correct solution in a finite time. Let us turn to the second class of problem in deductive and inductive logic and look for the possibility of an effective procedure of solution.

3.4 *Second problem: To examine a result.*

3.4.1 *Deductive logic.* Given a set of premises and a conclusion, the problem is to decide whether the conclusion is logically implied by the premises. That is to say, if a mathematician is given a set of axioms of geometry (called e) and an expression h, the problem is for him to determine whether h is a theorem, i.e., whether h is logically implied by e. For example, e might be the axioms of Encledean geometry and h might stand for the sentence "If two intersecting lines are straight, then the sum of the adjacent angles equals 180." This second problem is very closely related to the first problem (i.e., that of *finding* a conclusion), because once a mathematician has "found" an expression which he believes is a theorem, he must construct a formal proof in order to convince others that the expression in question is a theorem. Suppose a mathematician has found an expression h which he believes is a theorem, but assume that he cannot construct a proof. His negative result can mean either that h is not a theorem *or* that he is not sufficiently ingenious. In order to determine which of the above alternatives is true, we would need some mechanical procedure for deciding whether or not h is a theorem. However, it can be said that, in general, there is no effective procedure for deciding whether an expression h is implied by a set of premises e.

The problem of finding a group of rules which would allow one to decide mechanically whether an expression h is implied by some other expression e is called the "decision problem," and we know that only for a very small section of modern logic is there a decision procedure. In the Sentential calculus one can decide whether e implies h by forming the corresponding conditional proposition with the premises as the antecedent and the conclusion as the consequent, and testing the proposition for logical truth by means of a truth table analysis. Since there is a decision procedure for the Sentential calculus, a machine could automatically decide about logical implication and, in fact, such a machine has been constructed.[8] In addition to the Sentential calculus, there is a decision procedure for the Monadic Predicate calculus and a few other specialized areas of logic.[9] However,

[5] It is questionable as to whether the conclusion of an inductive inference is even probable. See Reichenbach, *op. cit.*

[6] I have used Carnap, *op. cit.*, as a source for large portions of sections 3.3 to 3.5 inclusive.

[7] A. Einstein, "On the Method of Theoretical Physics," Oxford Press, pp. 11-12; 1933.

[8] The author has built an electromechanical decision machine for the Sentential Calculus. It can handle up to eight input variables and can *decide* whether any given statement is implied by some given input information. (The machine is able to make this decision in approximately 30 seconds.)

[9] Tarski has given a proof that there is a decision procedure for elementary algebra. This means that we could build a decision machine for elementary algebra which would automatically decide whether any well-informed expression of elementary algebra is a theorem or not. See A. Tarski, "A Decision Method for Elementary Algebra and Geometry," Univ. California Press, Berkeley; 1951.

or quantification theory in general, and hence for all higher branches of mathematics, it has been proven that is impossible to find a decision procedure. If there were a decision procedure for arithmetic, one could feed into a computer such unsolved problems as Fermat's "last theorem", and the machine would be able to decide for us whether or not this famous conjecture be true or not.[10]

Thus, we find that for those areas where there is no decision procedure one must grope for a deduction that constitutes a proof of a theorem, and whether or not an individual does find the desired deduction is a matter of "luck" or ingenuity or intuition. Consequently, for this second problem we may state that, in general, (in deductive logic) there is no effective procedure for instructing an individual in which sequence to operate on the premises e in order to derive a particular theorem h. As Carnap has stated:

> "Constructing a proof is often called a rational procedure because here fixed rules have to be taken into consideration. However, the decisive point must not be overlooked: the rules of deduction are not rules of prescription, but rules of permission and prohibition. That is to say, the rules do not tell the logician X which step to take at a given point in the course of deduction; in other words, they do not constitute an effective procedure. The rules tell X merely which steps are permitted and thereby they say implicitly that all other steps are prohibited; they leave it to X to choose one of the steps permitted. Thus, here again, it depends upon X's ingenuity and luck whether he solves the problem, that is, whether he finds a series of steps permitted by the rules, such that they lead from e to h.[11]

3.4.2 *Inductive logic.* The corresponding problem in inductive logic is: Given some factual data e and an hypothesis h, now determine to what degree e confirms h. That is to say, if a physicist has some experimental data and has found a solution to problem 1 (above), i.e., he has found a theory to explain the observed facts, the second problem is to determine how good the theory is, i.e., to what degree is it warranted or confirmed by the observed data.

In general, there is no effective procedure for determining to what degree an hypothesis is confirmed by some given data. Consequently, we see that it is impossible to program a computing instrument to find a good hypothesis or to determine how good a given hypothesis is, on the basis of some given evidence.

3.5 *Third problem: To examine a given proof.*

3.5.1 *Deductive logic.* In order to explain this problem, let us assume that mathematician has found an expression h which he believes to be a theorem of logic or mathematics (problem 1), and also he has found what he believes is a proof that it is a theorem (problem 2). The third problem is whether there is an effective procedure for determining if an alleged proof is correct. Referring again to our explanation of what constitutes a proof, we see that a proof is a sequence of logical steps which start from the premises (initial information) and lead to the conclusion (output information) in such a way that each step in the sequence is justified by the fact that it does not violate any of the given transformation rules. With this notion of a proof, we see that there is a mechanical procedure for determining whether any sequence of expressions is, in fact, a proof. The procedure is merely to examine each step in the sequence and compare the alleged justification for that step with the list of transformation rules. Hence, one could program a computer so that it would determine whether an alleged proof were correct.

Given our basic notion of a computing machine as a language transducer, we see that a computer program corresponds (in some sense) to our notion of a proof. That is to say, a program is a sequence of orders which direct the machine to operate on given information according to certain transformation rules. The programmer must know what the sequence of orders in a given program will lead to the desired conclusion. Since there is a solution to problem 3 above, it would be possible to program a computer to check its own program.

3.5.2 *Inductive logic.* The corresponding problem in inductive logic would be the situation in which a scientist has found an hypothesis h and an alleged proof that h has a certain degree of confirmation on the basis of the observed data e. This third problem for inductive logic cannot be answered without obscurity due to the very questionable logical status of inductive logic, and since this problem has no immediate significance for the contents of this paper we shall ignore it.

3.6 *Summary (of section 3).* There are certain problems whose solutions can be found mechanically by following a set of fixed rules. Whenever one presents a proof in mathematics, we can check the proof by following a set of rules. Whenever there is a problem to be solved requiring computation only, we can solve the problem by following fixed rules of arithmetic. We might say that problems of this sort fall into the "context of justification and computation." That is to say, there are logical steps and rules that one may use in order to justify the grounds for a certain conclusion and logical rules that one can use in order to perform calculations. However, in order to obtain a good conclusion (theorem, hypothesis), one must "discover" it, and we shall say that this process of "finding" falls within the "context of dis-

[10] Fermat had asserted that he had a proof that the expression "$x^n + y^n = z^n$" cannot be solved for integer values of n which are greater than 2. Fermat's so-called proof has never been found, nor has anyone been able to prove or disprove this famous conjecture.

[11] Carnap, *op. cit.*, p. 195.

covery.'"[12] Whereas one uses all of the tools of logic in the context of justification – the context of discovery is "illogical." We *discover* by some vague, hidden, psychological means – not by rules. Thus, for one class of problem one has computers to obtain a solution. Let us now ask whether it is possible to use computing machinery as an aid in problems of discovery.

4. A MECHANICAL DISCOVERY DEVICE?

4.1 *Initial comments.* Since there is no effective procedure for discovery, we cannot program a computer to find hypotheses and also be *sure* that the output of the machine will constitute an *adequate* hypothesis. However, we wish to show that since physical theories in science are only tentative and always open to modification in the light of new experience, it is possible to devise a strategy for a machine by which it will formulate hypotheses and learn to modify the hypotheses on the basis of increased experience (i.e., input data). We shall have no guarantee that the hypotheses which the computer produces are completely adequate. Nevertheless, given some reasonable assumptions, we shall show that a machine can *learn* to modify its hypotheses and eventually formulate theories that will be good enough for making successful predictions.

4.2 *Physical laws in a very simple universe.* In order to clarify the notion of a physical law so that we may see whether a machine can discover such laws, let us discuss a simple universe (called "U") and the laws which describe the happenings in U. U is a finite and bounded universe, and it contains only two objects which are similar coins. Since the universe is so simple, it can have only the following describable states. Either the coins can be heads or tails, touching one another or not, and touching the edge of the universe or not. (Think of the universe as a board on which a game is being played.) We shall ignore the relation of time in U and shall use the following language (called "L") to talk about the possible states of this very simple universe. Our language L which is needed to talk about U shall contain two individual terms, a_1 and a_2, to name the two coins, and three predicates, P_1, P_2, and P_3, to denote the three possible properties that a_1 and a_2, may have. An atomic sentence in L shall consist of an individual term and a predicate term, and other molecular sentences may be formed from the atomic sentences by means of the usual sentential connectives (i.e., negation, conjunction, alternation, conditional and biconditional). The following is a list of the possible atomic sentences that can be formulated in L in order to talk about U.

1. P_1a_1 Coin 1 is a head.
2. P_1a_2 Coin 2 is a head.
3. P_2a_1 Coin 1 is touching edge of board.
4. P_2a_2 Coin 2 is touching edge of board.
5. P_3a_1 Coin 1 is touching coin 2.
6. P_3a_1 Coin 2 is touching coin 1.

(Since "P_3a_1" is equivalent to "P_3a_2", there are only five different atomic sentences.) Each of the above five different sentences or their negations can be combined (by the operation of conjunction) in 32 possible ways. The 32 possible states of this simple universe can be completely described by the following 32 state-descriptions.

1. $P_1a_1 . P_2 . P_1 . P_3 . a_2 . P_1 . a_2 . P_2a_2$
2. $P_1a_1 . P_2a_1 . P_3a_1 . P_1a_2 . \overline{P_2\ a_2}$
3. $P_1a_1 . P_2a_1 . P_3a_1 . \overline{P_1\ a_2} . P_2\ a_2$

.

.

.

.

32. $\overline{P_1\ a_1} . \overline{P_2\ a_1} . \overline{P_3\ a_1} . \overline{P_1\ a_2} . \overline{P_2\ a_2}$

(The symbol "." is used to denote the term "and".) Thus, for example, state-description 1 states that coin 1 is head and it is touching coin 2 and it is also touching the edge of the board and coin 2 is also head and touching the edge of the board. The other state-descriptions are read accordingly, with the line above an atomic sentence to indicate that it is negated.

It might be pointed out that since the five atomic sentences are binary statements (i.e., they are either true or else false), the state descriptions can be formulated in terms of a 5 bit binary number.

Of the 32 possible states of this "universe," only some of them can be actualized without violating the laws of U. The laws of U which describe those states that are "unallowable" are as follows:

1. The two coins cannot both be head. $(\overline{P_1\ a_1 . P_1\ a_2})$
2. If coin 1 is touching the edge of the universe, then either coin 2 cannot also be touching the universe or coins 1 and 2 must be touching one another. $(P_2\ a_1 \supset (\overline{P_2\ a_2} \vee P_3\ a_1))$
3. Both coins cannot be touching one another if coin 1 is touching the edge of the board and vice versa. $(\overline{P_3\ a_1} \equiv P_2\ a_1)$

(The symbols "v", "⊃", and "≡", stand for disjunction, conditional and biconditional, respectively.) The above rules imply that 23 of the 32 possible states cannot be actualized – hence, there are only 9 states that can ever occur. State-description, 5, 6, 7, 8, 9, 11, 21, 22, and 25 are the only ones that can possibly be true without violating the laws as stated above.

Physical laws are descriptions of those of the logically possible states of the universe that can actually occur. For example, there is a physical law which asserts that an unbalanced force acting on a body will cause it to accelerate. This means that (if the law is true) it is impossible to put an unbalanced force on a body and not

[12] Reichenbach, *op. cit.*, p. 7.

ave it accelerate – which is another way of saying that lthough it is *logically* possible for a body which is cted on by an unbalanced force to remain at rest (or in a tate of uniform motion), the above mentioned law states hat this possibility cannot be actualized in this universe. Likewise in U, it is logically possible for state-description 1 to be true, but the laws of U state that such a sitution cannot occur (because if it did it would "violate" he laws of U).

Suppose that the coins (a_1 and a_2) were placed in the niverse in various positions, but never in such a way hat the laws were falsified, and suppose that for each tate of U that was "allowable" we "feed" into a computer he corresponding (i.e., "true") state-description. Could he computer *discover* laws 1, 2 and 3, on the basis of a imited amount of data? Since there are no rules for discovery, what possible procedure could a machine follow n order to "find" the laws?

4.4 *A machine strategy for discovery?* If we think bout the mechanism that is involved in evolution, we see that it suggests a strategy which a computer might se to find the laws of U. There are two key mechanisms n evolution, and they are randomness and selectivity. In a sense, genetic structure is changed at random and the environment of the species acts as a selective element with the result that only the fit are "selected" to survive. The combined operation of randomness and selectivity eads to a result which has apparent order, viz., higher nd better adapted individuals.

The strategy that our computer shall use is that of formulating hypotheses at random (or systematically) and hen using the input data as a selective factor, and thus it will reject those hypotheses which are falsified on the basis of the input data. The computer will take the input data, i.e., one or more of the state-descriptions and using he five atomic sentences (stated in section 4.2), formuate arbitrary hypotheses using the atomic sentences as primitive symbols and the sentential connectives as primitive operations, and it will combine the symbols and operations in accord with the "built-in" formation rules n order to end up with well-formed expressions. Thus, in a sense, the computer will have solved problem 1 (section 3.3), i.e., by following the formation rules, it will generate "meaningful" hypotheses at random. The machine next must decide whether an hypothesis is false or not, and it can do this by comparing the hypothesis in question with the input data (which it has stored in the memory). As soon as an hypothesis is falsified, it is "dropped" by the machine and another is randomly generated. Thus, if the input data are governed by laws (i.e., by the laws of U), they will exhibit a certain uniformity, and after a finite number of state-descriptions are "fed" into the machine and using the method of randomness and selectivity, the computer will have formulated "good" hypotheses (i.e., in a sense it makes an inductive inference) and eventually it may find the laws (or a logically equivalent set).

We have no grounds for asserting that a computer will definitely discover the correct laws, even on the assumption that the universe, in question, has a finite number of states. Unless the machine knows all of the true state-descriptions, i.e., unless it has complete information concerning allowable plays, it cannot be *guaranteed* that the hypothesis under consideration will not be falsified with some additional input information. However, it is reasonable to assume that as the input is increased, and the machine *approaches* a state of complete information, the device will be making successful predictions about allowable states, before it definitely knows the laws of U.

5. CONCLUDING REMARKS

We have stated earlier that if one wants a guarantee that a machine will find a solution to a particular problem in a finite number of steps, then the problem must have an effective procedure of solution. Further, there is no effective procedure for discovering good hypotheses in empirical science, and hence one could not program a computer to find hypotheses *and be sure* that the output would represent *good* hypotheses. However, we also know that the nature of empirical science is such that one can never have certainty about the correctness of physical theories, and consequently any hypothesis that is discovered by a scientist is always open to modification in the light of new experience. Realizing the logical status of particular problems, it is unreasonable to ask a machine to do the impossible. Men cannot create theories that will never need modification, and the same holds for machines.

The actual psychological mechanism involved when men discover is not known, but the interesting thing is that the *result* of this mechanism (viz., discovery of hypotheses) can be copied by a machine. Using the strategy of randomness plus selectivity, a machine can also find good hypotheses, and consequently, although the operation of the machine is different from that of man the *result* can be essentially the same. This may help to clarify the question that has been raised many times in recent years, namely, "Can machines think?" Obviously, the answer to this question depends on the definition of the term "think," and it is a very difficult term to explicate. Nevertheless, we may attempt to answer the question in the following indirect way. We can say: "Show us some information manipulation of which a man is capable, and we can build (or program) a machine which will produce the same result. The actual *method* employed by the machine will be different, but the result will be essentially the same. Again, we do not know what the psychological mechanism is by means of which gifted scientists suddenly "hit upon" fruitful hypotheses, but clearly it is theoretically possible for a machine operating via a different *method* to arrive at the same *result*.

SYSTEM DESIGN OF THE SEAC AND DYSEAC

A. L. Leiner, W. A. Notz, J. L. Smith and A. Weinberger
National Bureau of Standards
Washington, D.C.

SUMMARY – In the course of developing the system plans for the DYSEAC and the SEAC, certain standard methods and procedures were evolved for producing a large-scale digital computer design. These standard procedures cover, first, the development of system specifications, second, the development of functional plans, and finally the development of wiring plans. The later stages of these procedures are reducible to sequences of simple steps, capable of being systematically formulated in explicit terms. The similarity between these procedures and many of the data-processing procedures commonly being executed by present-day computers suggests that, with further development of these design techniques, the wiring plans for new computer systems might well be produced by existing digital machines.

INTRODUCTION

This article discusses some of the factors which governed the choice of system features in the SEAC and DYSEAC and describes some of the standard procedures which were developed for working out their system designs.

The flow of development generally followed in creating such large-scale computers is charted in Fig. 1. As indicated, two sets of factors (which can be considered as the initial boundary conditions of the system-design problem) affect the choice of system features for a machine: first, the set of factors related to the intended use of the machine, and, second, those related to the type of components or "building blocks" with which the machine is to be constructed. Because these two sets of factors are basically unrelated to each other, they often

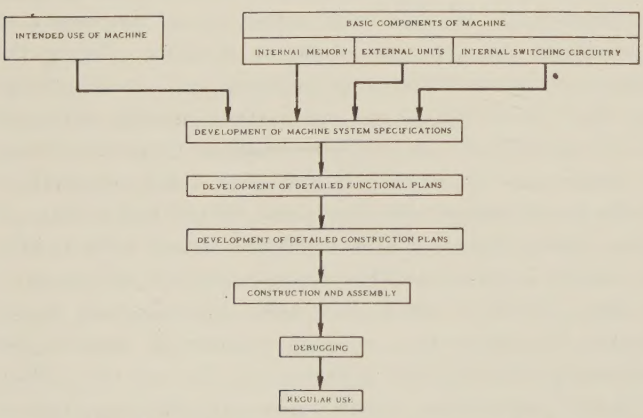


Fig. 1 – Computer development flow chart.

present contradictory requirements. For example, a proposed machine feature may appear ideal when evaluated solely in terms of the intended use of the machine but may entail an unacceptable engineering risk when evaluated in terms of component reliability or cost. The necessity for effecting compromises and avoiding conflicts of this kind between the rival claims of operational effectiveness and engineering reliability and economy strongly influenced the system design of the SEAC and DYSEAC.

As is also indicated in Fig. 1, the principal machine components whose properties profoundly influenced the system designs of these computers were the internal memory units, the external communication units, and the internal switching and small-scale storage circuitry. The internal memory units included an acoustic delay-line memory and an electrostatic Williams' tube memory. The external communication units included such devices as mechanical keyboard-printers, magnetic recording units, special cathode-ray tube display devices, input converters for translating analog information to digital form, and digitally-actuated output mechanisms. The internal high-speed switching and storage circuitry included the following fundamental digital elements for controlling pulse signals:

1. The *AND-gate*, with or without an inhibition input, and the *OR-gate* were the fundamental elements utilized for combining or switching pulse signals. Groups of these gates are assembled with an amplifying tube and pulse transformer to make a *pulse repeater*. This pulse repeater carries out the logical switching functions of the gates included in it and also amplifies and restandardizes the signals going through it.
2. The so-called *dynamic flip-flop* was the fundamental bistable device utilized for providing one-bit storage. This device is composed of a pulse repeater and a *delay line* connected to form a closed loop around which a single pulse can be kept circulating repeatedly with a re-circulation period of exactly one pulse-repetition cycle.

These elements were used uniformly throughout the SEAC and DYSEAC both for word processing and for central functional control purposes. No other basic elements were used in the internal system. In Fig. 2, items 1 through 6 illustrate these fundamental elements and the symbols adopted for representing them. Table 3 explains their mode of operation.

From these basic elements, small composite units were developed for carrying out typical simple processing operations according to the rules of binary arithmetic. For example, some comparators, counters, decoders, complementers, adders, storage and shifting registers that were developed are illustrated in Fig. 2, items 7 through 16. Using these small composite units, larger subsystems were then organized for carrying out more complex arithmetic and control operations such as the arithmetic operations of multiplication and division, or

control operation of selecting a word from a designated memory location. Table 4 lists some typical operations for which such subsystems were developed.

These composite units and subsystems provided a set of basic techniques by means of which computations could be performed on digital data and by means of which complex procedures could be employed for integrating large masses of unorganized information. Once devised, they served as a storehouse of building blocks and organization schemes from which more comprehensive full-scale systems could be developed. In this way, they provided the means for fashioning automatic supervisory control facilities capable of directing large families of external devices carrying out complex tasks.

DEVELOPMENT OF SYSTEM SPECIFICATIONS

In developing system specifications for the SEAC and DYSEAC, an effort was made to specify a balanced system in which each component part was organized to do only what it needed to do and no more. Such a system usually contains the fewest possible parts and consequently is more economical to construct, debug, and maintain. Since the characteristics of the principal memory, switching, and external communication units to be incorporated into the system were widely varied, the problem of achieving an effective balance between these units arose. A major boundary condition to the problem was imposed by the engineering decision to use a mercury acoustic delay-line memory for high-speed storage. The access-speed characteristic of this type of memory governed the choice of computing speeds for the switching units and input-output speeds for the external communication units. More specifically, a purely serial arithmetic unit was chosen instead of a serial-parallel or fully parallel unit, because of the relatively long average access delay (168 microseconds) imposed by the recirculation period (384 microseconds). Since each of the recirculating tanks contains eight words 48 binary digits long (45 information digits plus 3 spaces between words), the total recirculation time is 384 microseconds. If the eight words in each tank are randomly arranged, therefore, the average time required for some particular word to become accessible is 7/16 of the total period, which is 168 microseconds.[1] For the four references to the memory required in most SEAC and DYSEAC arithmetic operations, this lengthens the time needed to execute an operation by 672 microseconds. Since the actual basic computing time required to carry out the four sequential steps of a complete addition operation using simple and efficient serial techniques is only 192 microseconds, obviously not much over-all gain in speed would be achieved by reducing the arithmetic computing time unless a corresponding reduction could be effected in the memory access time. Table 1 shows the average times for execution of the various types of arithmetic operations performed by SEAC and DYSEAC and the portion of these times occupied by memory access waits.

Table 1

Average Rates for Certain SEAC and DYSEAC Operations

Operation	Performance Rate in SEAC (milliseconds)	Performance Rate in DYSEAC (milliseconds)	Percentage of Time Occupied by Memory Access Waits
Addition, Subtraction, Logical transfer	0.9	0.9	78%
Multiplication, Division	3.0	3.0	23%
Comparison	0.7	0.7	72%
Shift (halfword shift)	–	1.1	61%
Justify (halfword shift)	–	2.0	33%
Summation (per word, for 100 words)	–	0.06	12%

NOTE: Each word contains 45 binary digits. (44 numerical digits plus 1 sign digit).

From this table it will be noted that addition times and multiplication times are the same for DYSEAC as for SEAC – the speed ratio being about 3:1 for executing these operations. If, as is the case in many computational problems, additions occur about three times as often as multiplications, the speed-up achieved in the over-all execution time of a problem by striking out a given percentage of the operations in the program is the same regardless of whether the operations eliminated are additions or multiplications, or a mixture of both. In this rough sense, the speed ratios may be said to be balanced. Actually, the occurrence of multiplication in solving a problem on DYSEAC will be rather less than with SEAC because of the newly added high-speed Shift and Justify operations by which many procedures requiring one complete multiplication time with SEAC can be done much more rapidly with DYSEAC.

Turning now to the balance between computing rates and input-output rates, Table 2 shows the time required to transfer a word between the high-speed memory and an external unit operating at the pulse-repetition rates associated with conventional magnetic recording units. Since the time required to execute an average instruction is of the order of one millisecond, it takes four to six times as long to transfer a given instruction word from a typical external magnetic unit into the high-speed memory as it

[1] L. Leiner, "Buffering Between Input-Output and the Computer; Review of Input and Output Equipment Used on Computing Systems," Joint AIEE-IRE-ACM Computer Conference, 1953, pp. 22-31; March, 1953.

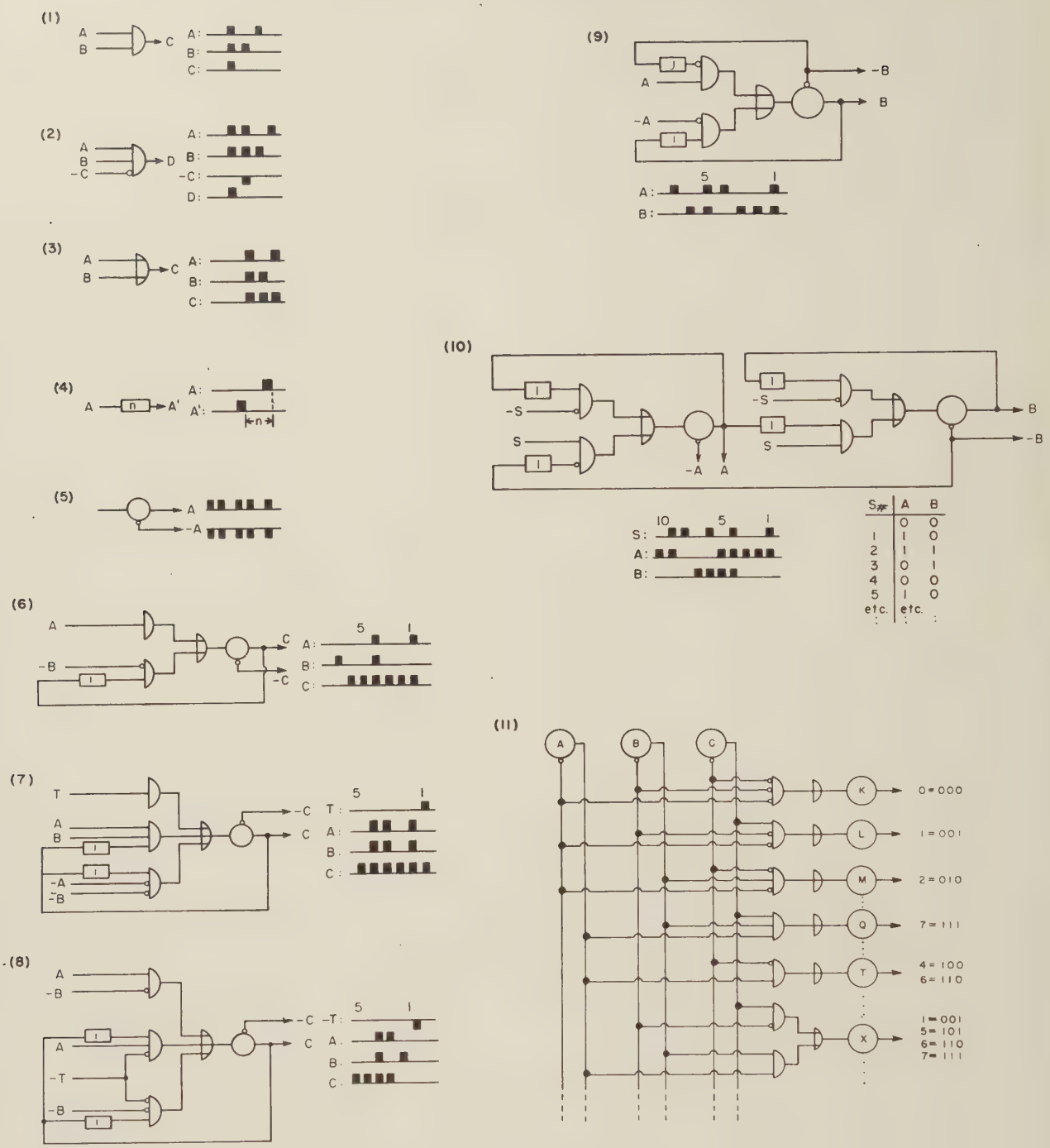


Fig. 2(a) – Fundamental elements for computer systems (in two parts).

takes to execute it after it has been stored inside the machine. Ordinarily, associated with each instruction there are from one to four words of input-output, namely, at least the instruction word itself and possibly any or all of the operands and the result. Therefore, the total input-output time required to transmit these data in and out of the machine would not ordinarily be greater than roughly two dozen instruction-execution times. If each instruction is executed about this many times after it has been inserted into the machine, then the internal processing and external transfers will, in the long run, occupy roughly the same amount of time. In the case of DYSEAC, where computing and input-output transfers can proceed concurrently, the over-all time taken to solve a problem will be equal to the time required for the longer of these two processes.

It appears, therefore, that for a high-speed memory of the present acoustic delay-line type, little benefit would accrue from increasing these input-output rates unless the problem being solved involves extensive

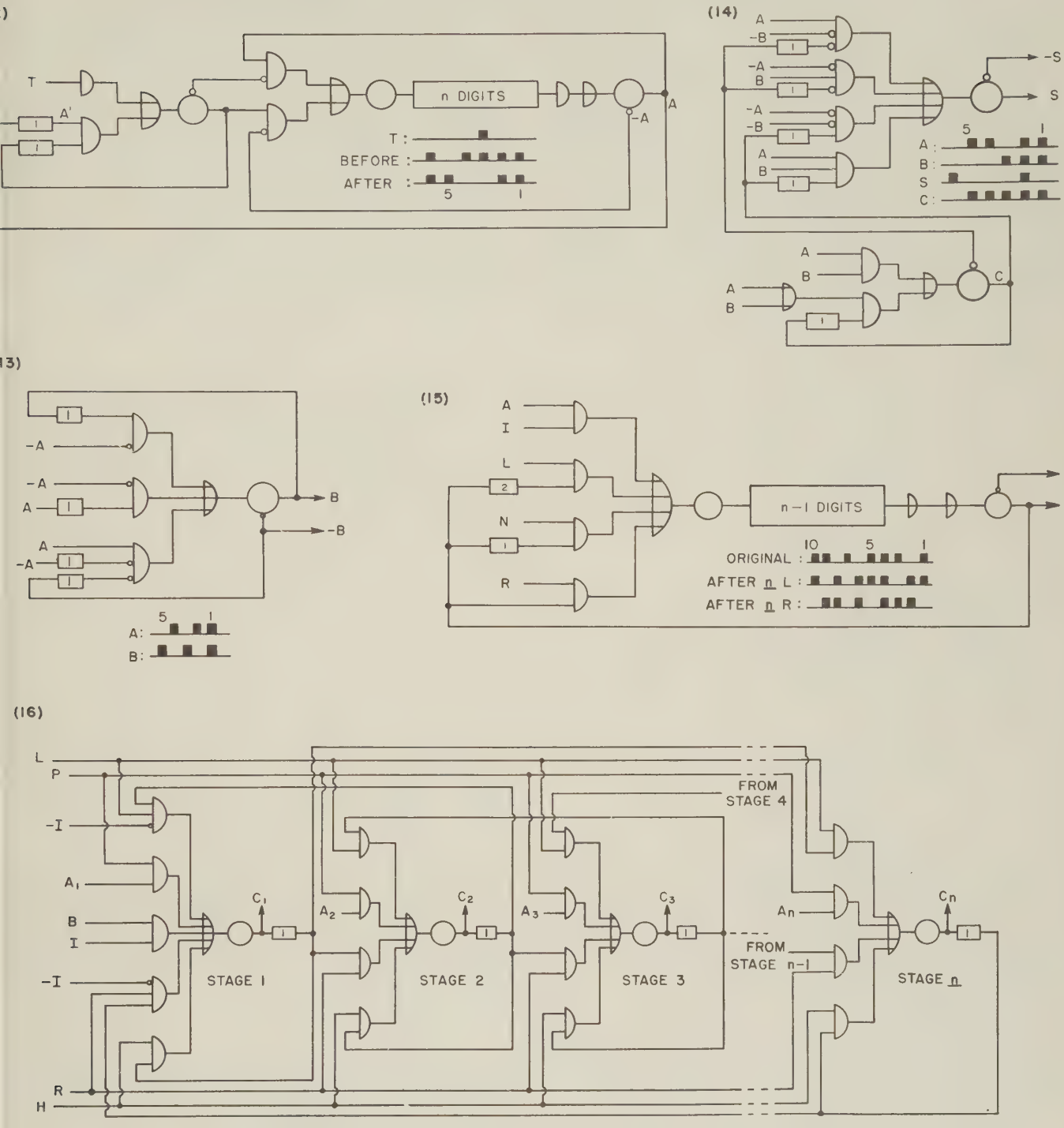


Fig. 2(b) – Fundamental elements for computer systems (in two parts).

ing along the wire or tape in search of particular ds. If the nature of the application requires such ing procedures, then a better-balanced system could obtained by increasing the over-all input-output rate irect proportion to the average number of words which t be searched through in order to find a desired word.

Other major problems were encountered in formulat- system specifications for the SEAC. On the one hand e specifications had to be kept to a minimum so that machine could be completed at the earliest possible date for interim use, and on the other hand the specifications had to include provision for expanding this interim machine, both internally and externally, by the annexation of advanced high-speed internal memory equipment and external input-output equipment as they became available. Neither the developmental time-scale nor the operating properties of these new components could be precisely defined at the time the specifications had to be drawn up. Furthermore, the future uses of the machine could not be precisely defined either.

Fortunately, at the time the development of the DYSEAC was undertaken, the situation was quite different. Even though the range of application and method of use intended for the DYSEAC are broader and more varied than for SEAC, it was possible to define them more precisely at the start of the program. Consequently, the DYSEAC system could be organized more effectively than the SEAC; and though both machines contain the same relative proportions of equipment for carrying out communication, control, and processing functions (Fig. 3), the DYSEAC is able to provide considerably more powerful operating features using the same amount of equipment.

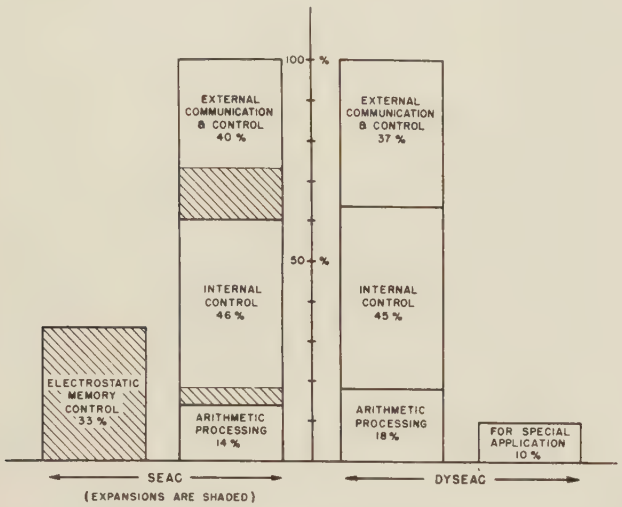


Fig. 3 – Proportion of equipment for performing major functions in SEAC and DYSEAC.

After the problems related to the uses of the machine and to the operating characteristics of the principal component parts had been defined and evaluated, and before the final system specifications were formulated, several other analytical studies were carried out. For example, coding studies were conducted to determine optimum internal programming specifications, i.e., what the format and content of the number and instruction words should be, what the program-sequencing procedures should be, what arithmetic and control operations should be included in the system, and other related questions. As a result of such studies, the approximate number of instructions and memory cells needed for solving various problems could be established, as well as the frequency of access to various classes of words, over-all solution times, etc. These data were in turn useful for carrying out further analyses aimed at determining the relationship between the problem-solving capacity of the system and the equipment cost. For example, by relating the unit cost and access time of various types of storage equipment with the relative frequency of use of the various classes of words involved in the program, the most economical choice of proportions for each type of storage device needed to perform the problem at a given over-all speed could be determined.

It should be noted that a strong interdependence exists between decisions made as a result of these various considerations. In designing SEAC, for example, the decision to provide for the incorporation of an additional parallel memory as well as the initial serial memory exerted a profound effect on the design of two apparently unrelated parts of the system, namely, the input-output buffering system and the program-sequencing system. The parallel memory was coupled to the otherwise serial machine by means of a staticizing shift register capable of communicating with the parallel memory in a simultaneous broadside fashion from all of its register cells and

Table 2.

Rates for Input-Output Operations With Magnetic Recording Units, Averaged Over 512 Words	
Total Pulse Rate of External Wire or Tape Unit (kilocycles)	Time to Transfer One Word Between External Unit and DYSEAC Memory (milliseconds)
4	12.8
8	6.8
12	4.8
16	3.8

with the serial units in a step-by-step serial fashion through a single register cell. The flexible serial-parallel conversion abilities of this device, together with its ability to operate over a wide range of shift speeds, led to its adoption as the input-output pick-up register for the system.[1] On the other hand, the DYSEAC, which does not require this sort of facility for staticizing information, uses the more economical circulating electrical delay-line storage technique for its pick-up register. This register, although purely serial in operation, can be loaded with up to 45 binary digits delivered from an external input device via 45 distinct channels in a broadside fashion.

A second effect of the decision to provide for a parallel memory on SEAC relates to the program-sequencing system. Initially, the memory capacity of the machine was less than 1024 words, and therefore address numbers of 10 bits each were adequate. Hence, an instruction word containing four addresses of 10 bits each could be used. When it became necessary to provide for expansion of the initial memory capacity up to as much as 4096 words, extra space was needed in the instruction word to represent numbers in excess of 1024. To make room for the longer address code designation, therefore, without extending the length of the instruction word, a three-address instruction word of 12 bits per address was adopted as an optional alternative to the four-address instruction word. Along with this three-address designation, a consecutive-number scheme for sequencing instructions was used. This scheme in turn facilitated the

Table 3. Explanation of Fundamental Elements for Computer Systems Shown in Fig. 2

Individual pulses occurring successively in a pulse train are shown pictorially in the order right to left. A pulse represents a 1-digit; the absence of a pulse represents a 0-digit.

Item on Figure 2	Name	Mode of Operation
1	*AND-gate*	A pulse appears at C if and only if pulses occur on A and B simultaneously. More than two inputs may be used.
2	*AND-gate* with inhibition input	A pulse appears at D if and only if pulses occur on A and B simultaneously and no pulse occurs on C.
3	*OR-gate*	A pulse appears at C if a pulse occurs on either A or B, or both. More than two inputs may be used.
4	Delay-line	A pulse occurring at A appears at A' after the span of time n microseconds.
5	Amplifying tube and two-pole transformer output	The output at A consists of positive-going pulses; at -A, negative-going pulses. There is always a one-to-one correspondence between A and -A.
6	Basic flip-flop	The flip-flop is turned *on* (i.e., emits continuous pulses at C) beginning with the first pulse of A, and is turned *off* (emits no pulses) beginning with the first pulse of B which is not accompanied by a pulse of A.
7	Equality comparator	T is a priming pulse occurring before the first pulses on A and B that are to be compared. After the last pulses of the trains A and B have appeared, C emits a pulse if and only if pulse trains A and B are identical.
8	Inequality comparator	T is a clearing pulse that occurs at or before the first pulses on A and B that are to be compared. After the last pulses of A and B have appeared, C emits a pulse if and only if the binary number represented by A is greater than that represented by B. The digits of A and B occur in order of increasing significance.
9	Binary counter	B emits continuous pulses beginning when an odd number of pulses have occurred on A, but B emits no pulses when an even number of pulses have occurred on A.
10	Two-stage cyclic counter	Stepping pulses occur on S. With both stages off, the first S turns on A. If A is on and B is off, the next S turns on B. If A and B are on, the next S turns off A. If A is off and B is on, the next S turns off B, etc.
11	Matrix Decoder	Let the pulses (or no pulses) emitted by A, B, and C at any instant represent a three-binary-digit number, ABC. Examples of decoding are: M emits a pulse if and only if ABC = 2; T emits a pulse if ABC = 4 or 6. Such matrices are readily expandable so as to decode larger numbers of digits and in a wider variety of ways.
12	Add-1 counter	T is a priming pulse whose timing determines the power of 2 to be added to the n-binary-digit number contained in the delay-line register. Starting at the time of T, all digits of the original pulse train are inverted (1 becomes 0, 0 becomes 1) up to and including the first 0. The digits occur in order of increasing significance. If A' is from -A instead of A, the device subtracts 1.
13	Two's complementer	All digits of A up to and including the first 1 emerge from B unchanged, but all succeeding digits of A after the first 1 are inverted. Digits occur in the order of increasing significance.
14	Adder	A and B carry pulse trains that represent two binary numbers to be added. The output S is the pulse train that represents the sum of A and B. The carry digits are at C. All digits occur in order of increasing significance.
15	Recirculating delay-line register	This illustrates how an n-binary-digit number contained in such a register may be shifted to the left or right by precession. N, L, R, and I are mutually exclusive control pulses, and in the absence of all others, N is present. N is *normal recirculation*, L is *Precess left*, R is *precess right*, and I is *insert*. After a continuous train of n L- or R-pulses, the contents have been shifted one pulse position to the left or right with respect to their original positions. Apart from mere shifting, such a precessing register may be used as a pick-up register to collect digits presented on input A at a synchronous but irregular rate, in normal or reverse order, and to arrange them in normal order for distribution as a pulse train. This is done by causing a single I-pulse corresponding in time to each A-digit to be substituted for the first pulse of a train, of L-pulses or for the last pulse of a train of R-pulses.
16	Staticizer	This illustrates a different type of pick-up register, and it also shows a method of converting from parallel to serial mode of transfer and vice versa. I is serial-insert control pulses for source B. P is parallel-insert control pulses for sources A_1, A_2, A_3, ..., A_n, H is *hold* pulses, R is *shift right*, and L is *shift left*. I, P, H, R, and L are mutually exclusive except that I is always accompanied by R or L. In the absence of all others, H is present. A single R-pulse causes the content of each flip-flop to be transferred to the one immediately to the right, and a single L-pulse, to the left. For normal order of B-digits, a single R-pulse is used with each I-pulse, while for reverse order of B-digits a single L-pulse is used instead. A continuous train of R-pulses allows an information pulse train to be inserted or removed from the register in normal serial order. A single P-pulse inserts digits from A_1, A_2, A_3, ..., A_n simultaneously. The parallel outputs are C_1, C_2, C_3, ..., C_n.

adoption of a floating relative-address scheme for program sequencing.[2] These examples illustrate how design decisions affecting a single unit of the system were propagated and made themselves felt throughout the remainder of the system.

DEVELOPMENT OF DETAILED FUNCTIONAL PLANS

The previous decisions defining the over-all system specifications to a great extent implicitly determined the general nature of the over-all blocks or major units in the system as well as their control inter-relationships and principal information-transfer routes. At this stage of development, the system is usually represented by the familiar block diagram composed of simple labelled boxes interconnected by means of directional lines (see Fig. 4).

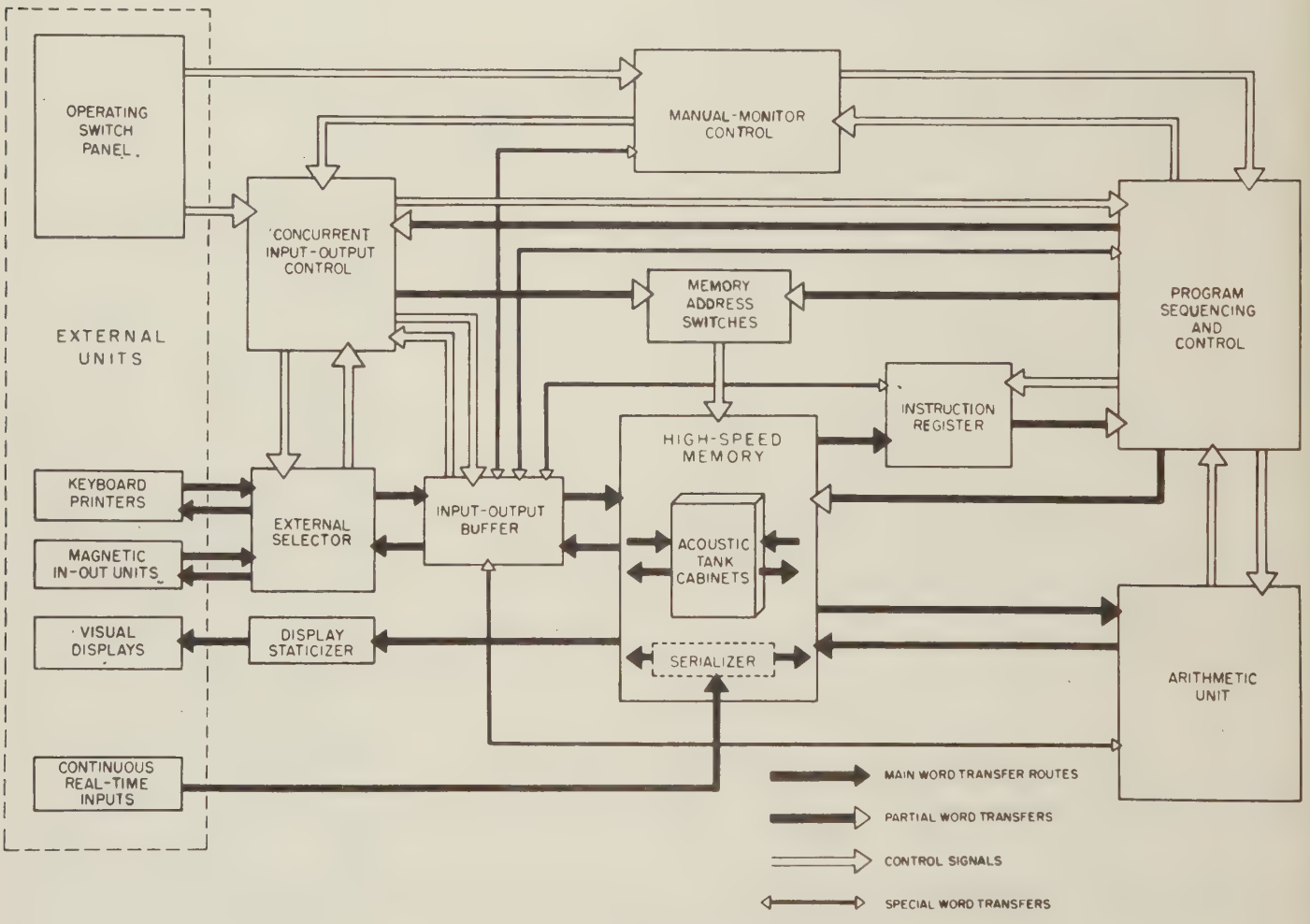


Fig. 4 – Block diagram of DYSEAC.

The next step in the evolution of the system design was to break each of these large blocks into smaller blocks, according to the functions suggested by the general operating specifications. For example, an arithmetic unit might be broken into boxes representing an adder and some storage registers whose number and characteristics depended on the operating specifications. A control unit might be broken into boxes representing subunits whose functions are, for example, the selection, acquisition, and distribution of information from the memory to other parts of the computer, or the issuing of control signals to the arithmetic unit that direct it through certain motions which are required in order to perform a given arithmetic operation. At this stage of development, definite rules regarding the cause-effect relationships among all the basic units became fixed. Also, the time delays imposed by the information processing, that is, the speeds at which signals can make their way through the various blocks, had to be estimated, and upper and lower limits specified.

Because the system being designed was centrally synchronous, over-all timing considerations now came to the fore. To use a rough analogy, all the units and subunits had to be so designed that they could be geared together and perfectly meshed. The next step, therefore, was to investigate and specify definitively the exact timing inter-relationships which must exist among all of the major units in order to insure that transfers of data

[2] A. L. Leiner, "Provision for Expansion in the SEAC, Mathemathical Tables and Other Aids to Computation," vol. 5, No. 36, pp. 233-237; October, 1951.

nd control signals among them can take place in proper equence. This step is necessary in order to force the *me* sequence of the passage of signals from one unit to

Table 4.

Examples of Typical Basic Operations in Computing Systems

Types of Operations	Procedures Which Must Be Developed for Carrying Them Out
Arithmetic:	
Addition Subtraction	Basic addition-subtraction procedures including negative number representation, complementing practices, sign determination of sums, overflow recognition.
Multiplication Division	Basic multiplication-division procedures including the shifting of numbers in a register, the flow of information between various registers, time sequencing and control of steps required in performing such operations.
Compound operations	Specialized procedures, formed from combinations of basic procedures, which are needed to achieve higher speeds of operation for special purposes, e.g., extra-rapid accumulation achieved by the DYSEAC Summation operation at up to nine times the normal DYSEAC Addition range.
. *Control:*	
Central control operations	Basic procedures for achieving centralized control functions. (In SEAC and DYSEAC, most control functions are performed by means of suitable combinations of the outputs of only four basic units: a timing generator that provides cyclic recurrent signals for marking off elapsed time within each serial word-cycle, a four-state phase generator whose outputs correspond to each of the four phases of each operation, a phase-stepping unit that advances the phase-generator and marks the initial cycle of each phase, and an operations generator that specifies the type of operation being performed.)
Memory-address selection	Procedures for achieving rapid memory access by means of space-selection switches using matrix decoders for digital address codes, time-selection using counter-timers, and space-voltage analog selection using special digital-to-analog converters.
. *External transfer operations*	Procedures for transferring digital information between internal memory of a computer and low-speed nonsynchronous input-output devices.[1]
. *External control operations*	Procedures for providing flexible external control over the internal system and for achieving versatile joint control by both internal and external parts of the system acting jointly.
. *Program sequencing*	Procedures for sequencing the individual instructions in a program and providing for branching in its execution.[2]

he next to correspond to the desired *cause-effect* sequence through which the units are functionally related to each other. This process is sometimes referred to as "closing the timing loops." For example, with a synchronous machine using a circulating delay-line memory, one of the major timing loops to be closed is the so-called memory-to-arithmetic-unit-to-memory loop. Closing this loop means establishing that a pulse read out of a memory location into the arithmetic unit can pass through the arithmetic unit and arrive back at the memory in time to be rewritten into the memory at exactly the proper instant. The proper instant for this rewriting is determined by the requirement that, when the pulse is read out of the memory at a later time, it will emerge from the memory at an instant which is precisely an integral number of word-transfer cycles after the instant at which it emerged on the previous occasion.

Once the major timing loops were closed, it became possible to establish a fixed over-all timing schedule for the system as a whole. That is to say, a definitive pulse-time "time-table" was adopted for most of the major units which communicate with each other and for the over-all timing-pulse generators whose functions are to furnish signals commonly used by all the other units. From this point on, every remaining unit and subunit in the system had to be so designed that it would be capable of performing its function entirely within the confines of the fixed time schedules adopted for the over-all timing signals.

It is advantageous to establish these fixed over-all time schedules at an early stage of the design program, because this facilitates dividing up the detailed work on the various major units among a team of several designers. So long as each designer adheres to the established timing "boundary conditions," the detailed design of each unit can be carried ahead separately with less fear of drastic reaction from developments in the design of other units. This sharing of design responsibility is feasible, however, only if the timing schedules are fixed at the outset and left unchanged.

In carrying ahead the design of each unit and subunit, the next step was to convert the logical cause-effect relationships which must exist within and among the various subunits into space-time relationships. The cause-effect relationships may be expressed by well-known algorithms for the case of an arithmetic unit, while for a control unit the description of the relationships may take the form of specially devised tables of procedural rules and process flow diagrams. Whatever their form, these cause-effect relationships may be reduced to stylized logical statements; for this purpose the notation of Boolean algebra often provides a convenient shorthand. In order to convert the cause-effect relationships to space-time relationships it was necessary, first, to select fundamental building blocks capable of executing the required functional relationships on incoming pulse signals, and, second, to determine the proper time at which these signals should be transmitted between the individual inputs and outputs of each building block.

This work was carried out with the aid of *process block diagrams*, an example of which is shown in Fig. 5. These diagrams consist mainly of simple interconnected rectangular blocks labelled with descriptive titles designating the general logical relationships they bear towards each other. Each block represents a specific type of subunit drawn from a list of semi-standardized prototype units designed to perform various common functions. Typical examples of such units are the comparators, counters, adders, and registers described in the of Fig. 2. Although basically only five varieties are used, these are sufficient to determine implicitly every physical component of the machine. The internal details of each block on the process block diagrams were worked out in terms of these symbols and recorded on the functional diagram (Fig. 6). As noted above, this procedure consists largely of expanding, modifying, and combining these fundamental elements so as to fit them to the special case at hand. The manipulative techniques of Boolean algebra were occasionally found helpful at this

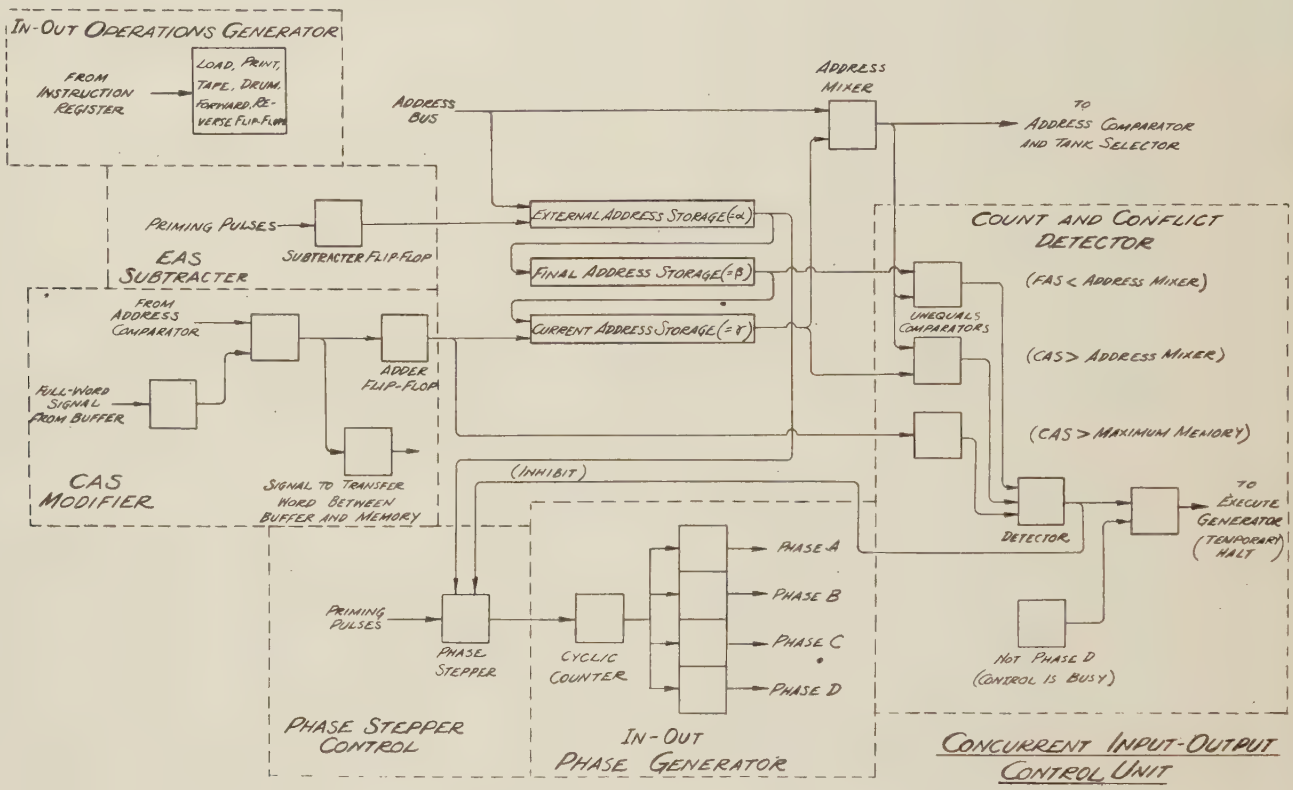


Fig. 5 – Example of a process block diagram.

introduction. From this collection of prototypes, a functional unit specially suited to any particular situation could usually be devised merely by making simple modifications of the standard prototype model.

The causal manner in which each building block fitted into a unit to perform its system function was further specified by prescribing the times at which it was to be turned on or off by each connected block. At first, the entire unit was laid out in blocks with timing signals indicated only approximately. As the design process proceeded, the necessary timing specifications became more precisely definable, and the specific timing pulses and exact delay-line lengths needed in order to insure proper time meshing of all the gating stages could be specified.

The next step, which overlapped the previous one to some extent, was to prepare so-called *functional diagrams* using the symbols illustrated in items 1 through 5 point for reducing gating configurations to standard form or for checking the equivalence of alternative circuit arrangements. Each tube stage was assigned a unique designation (usually a three-letter symbol), which serves to identify the signals that the stage emits. As the stages became meshed together, numbers specifying the entry and exit times of the signals for each gate, delay line, and tube were recorded. In designing SEAC, tolerances on these quantities were calculated nominally to the nearest .01 microsecond. With DYSEAC, however, because all components were standardized, it was sufficiently precise to record the timing on the functional diagram only to the nearest nominal quarter of a microsecond.

In the process of preparing the functional design, an effort was made to minimize the number of components utilized, particularly with respect to tubes, next with respect to delay lines, and finally with respect to diodes. The choice of the most economical and efficient arrange-

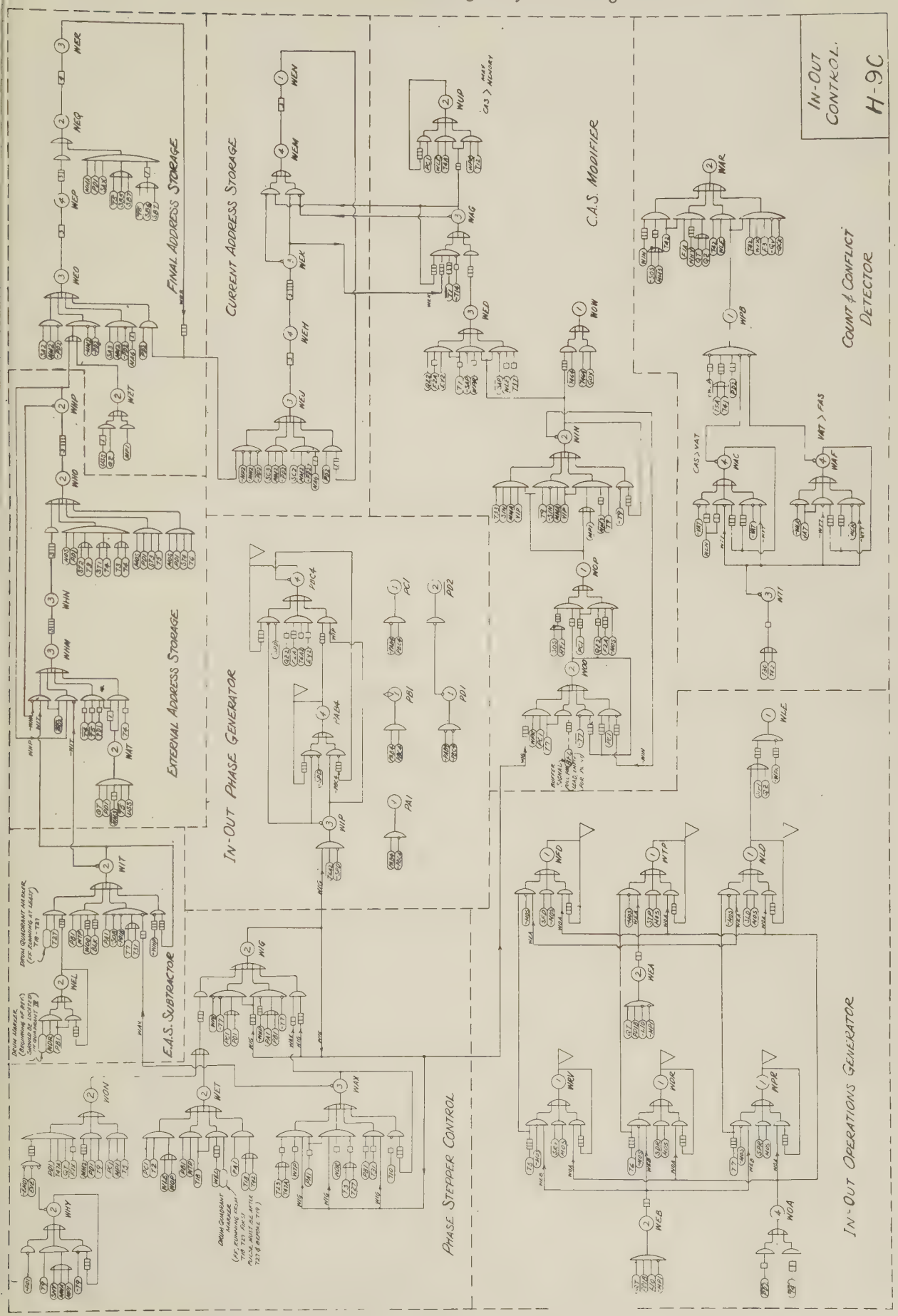


Fig. 6 – Example of a functional diagram.

ment for the functional elements depended quite strongly, however, on such factors as the size and content of the physical packages into which the electronic components were grouped and on their electronic operating specifications. Factors which affected the choice of optimum forms on the functional layouts were, for example, (1) the maximum permissible stray capacitance that is introduced as a result of extended lead lengths, and (2) the limitations on the maximum load that can be drawn from a tube. Functional layouts, therefore, even though they depict no electronic components explicitly, still must be designed with careful regard for the specific physical components by which they are later to be realized.

Another type of interaction between successive stages of the development program may occur after the functional design is partially or even fully completed. At this stage the designer may find that the operating capabilities of the machine can be usefully expanded by making minor design rearrangements which team up the same facilities in new combinations. In this way, additional operating features can often be provided at little or no additional equipment cost.[3] This "feedback" from functional design to system specifications can produce significant over-all improvements when close coordination is maintained between the two activities.

TRANSLATION TO DETAILED WIRING PLANS

After the functional plans had been completed, the evolution of the design proceeded toward the preparation of the detailed working plans from which the actual electrical wiring of components could be carried out. The form that these wiring plans took for each machine depended strongly on the physical packaging methods employed for holding the electronic components on the chassis and on the fabrication procedures followed in wiring up the soldered connections. Although extensive use was made of plug-in component packages in both SEAC and DYSEAC, the nature of the packages and the fabrication techniques used in both cases were so dissimilar that the types of wiring plans prepared for the two machines had to be radically different.

In the SEAC, since fabrication and wiring were carried out by skilled technicians who were capable of reading circuit diagrams, it was possible to use conventional wiring diagrams containing the usual electronic symbols for tubes, transformers, delay lines, resistors, and germanium diodes. In this machine, transformers and electrical delay lines up to two microseconds were packaged in individual plug-in units, but diodes were packaged in groups, electrically isolated from each other and usable in logically unrelated circuits. Over 20 distinct types of these diode packages were used, each package containing generally from 4 to 7 diodes connected in various standard configurations. After the circuit diagram for each chassis was completed, therefore, the diodes indicated on it had to be put through a so-called "clusterizing" process in which groups of diodes on the diagram were classified according to the different types of standard configurations to which they corresponded. Each such group was then surrounded by boundary lines and the enclosed area, designated according to its type, was assigned a socket location on the chassis. This clusterized circuit diagram was used by the wiring technicians in wiring up the chassis.

In the DYSEAC, however, a completely different form of component packaging was employed. (Each DYSEAC etched-plate tube package contains not only a tube-transformer unit but also preassembled AND-gates and OR-gates which can be associated with the tube. Also included are supplementary free components by means of which these standard preassembled gates can be expanded or additional gates incorporated.[4]) Furthermore, a considerable portion of the actual chassis wiring work on the DYSEAC was carried out by relatively inexperienced technicians who were not expected to cope with the type of circuit diagrams used for SEAC. As a result, sets of working plans were drawn up for DYSEAC which explicitly listed the precise socket and pin identification numbers of each of the approximately 10,000 pairs of socket pins in the machine between which soldered connections were to be made. In order to produce these, a method was employed which involved the preparation of three distinct types of working plans, namely *packaging diagrams*, *chassis charts*, and *wiring tables*.

The packaging diagram was prepared using the functional diagram as a guide. For each tube stage appearing on the functional diagram, a rectangular package symbol was entered on the packaging diagram (Fig. 7). The next step was to transcribe to the *inside* of the package symbol all of the gating appearing on the functional diagram which fell within the category of standard built-in types preassembled inside the package. Following this, the non-standard gating structures which were not of the preassembled types (that is, types which required the use of supplementary free diodes or other components) were transcribed from the functional diagram and entered *outside* the package symbols. Finally, components and connections required for circuitry reasons and not appearing on the functional diagram were entered. Thus the information on the packaging diagram is more complete than on the functional diagram in that it represents all the connections which must be made on the chassis sockets.

By means of the packaging diagram it was possible to determine the total number of packages that were required for all the gating stages and delay lines. After

[3] A. L. Leiner and S. N. Alexander, "System Organization of the DYSEAC, Trans. IRE-PGEC, vol. EC-3, no. 1; March, 1954.

[4] R. D. Elbourn and R. P. Witt, Dynamic Circuit Techniques used in SEAC and DYSEAC, Trans. IRE-PGEC, vol. EC-2, No. 1; March 1953.

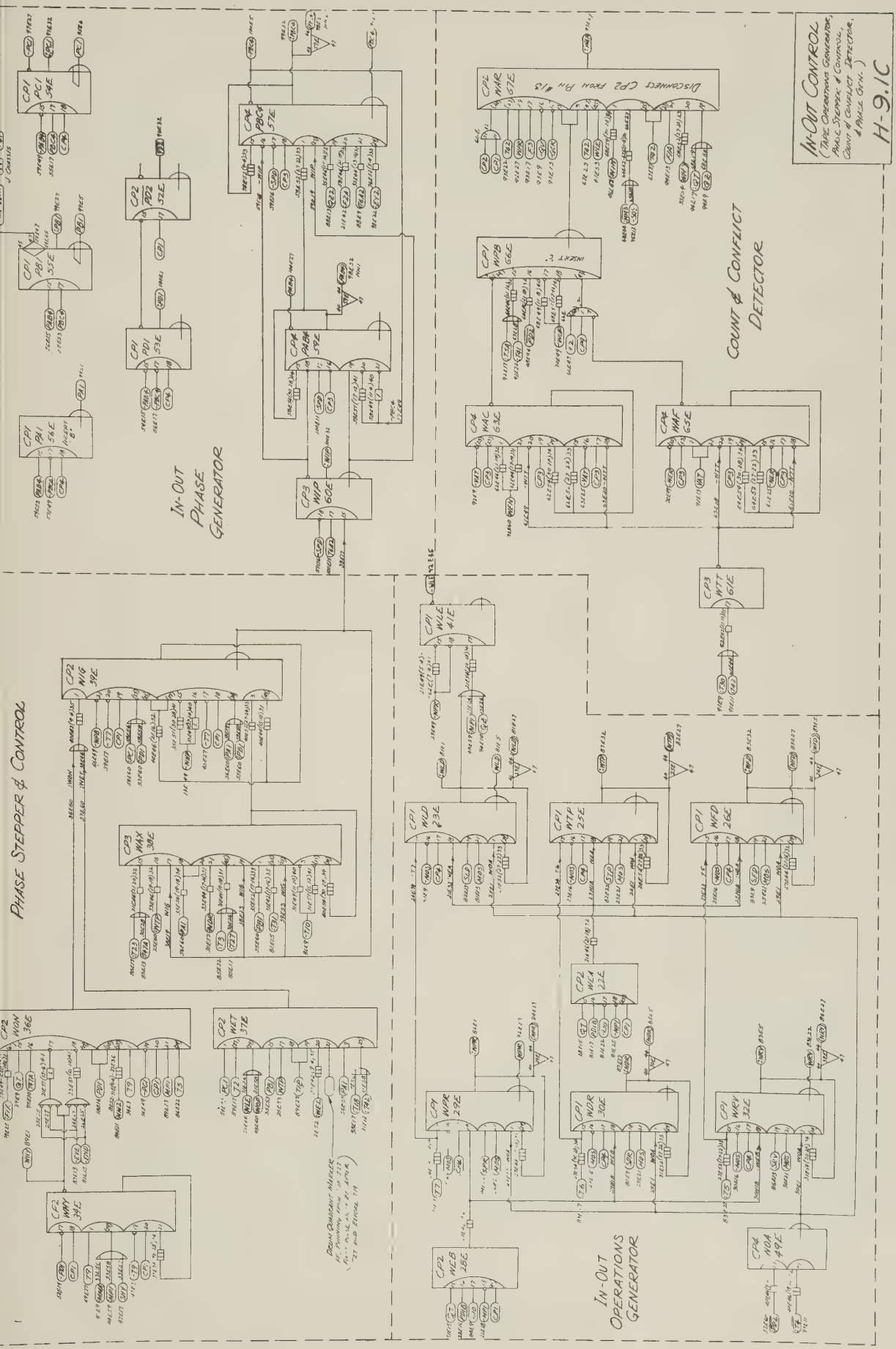


Fig. 7 – Example of a packaging diagram.

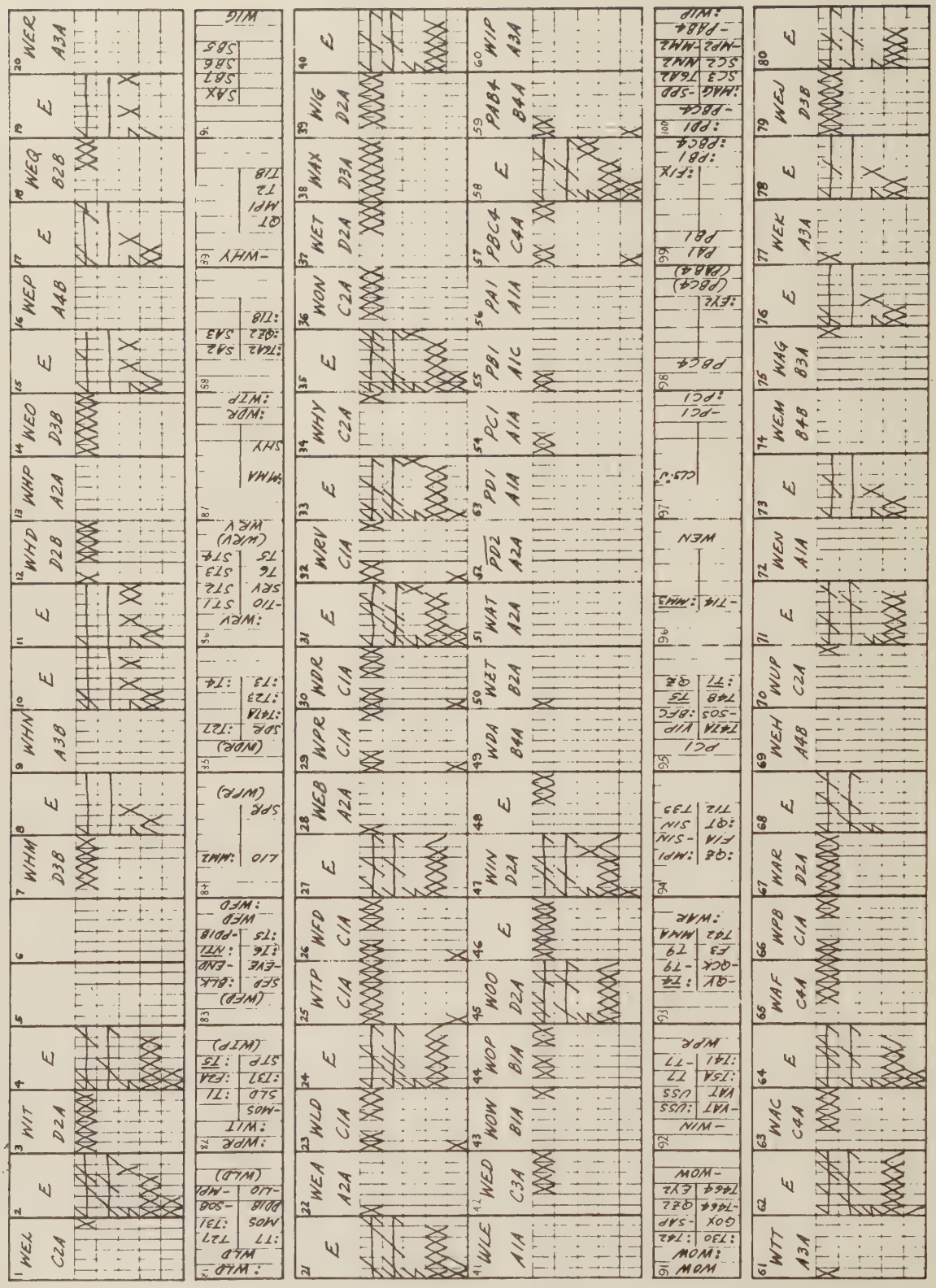


Fig. 8 – Example of a chassis chart.

25E TUBE		WTP CIA		26E TUBE		WFD CIA		27E LINE				28E TUBE		WEB A2A	
26D1	23D1		1	29D1	25D1						1				
(J30)			2	(J30)							2				
			3								3				
			4					46C7			4	(J44)			
			5					11C53			5				
			6					(35)			6				
			7					( )			7				
			8								8				
			9					(C43)			9				
			10					(C31)			10				
			11					( )			11				
			12								12				
			13					( )			13				
			14								14				
27H34			15	27H33				(C45)			15	51D15	22D15		
23F16	26F16	44F21	16	25F16	29F16			(C32)			16	22D16			
NC4			17	NC4				( )			17	84D9			
23H18	26H18		18	25H18							18	22F18			
82D22			19	83D9				( )			19				
			20								20				
23D21	26D21		21	25D21	29D21			(C47)			21				
			22					(T33)			22				
24C9			23					( )			23				
95D14			24								24				
83F15	26F25		25	25F25				( )			25				
(F28)			26	(F28)							26				
83F13	26F25		27	25F27				(T53)			27				
(F26)	21F50		28	(F26)	23F50			(T34)			28				
24H33			29	27H32							29				
(J2)			30	(J2)				(T55)			30				
			31					(C10)	29H29		31				
			32					(C16)	26H29		32				
			33					(T22)	26H15	32H15	33				
			34					(T28)	25H15	30H15	34				
			35					(6)	29H18		35				
			36					31C4	36H23		36				
			37								37				
			38								38				
			39	(L45)							39				
(L45)			40	(C52)							40				
(C52)			41								41				
			42					28C60			42				
			43					(C9)			43				
82L27	NN4		44	83L5	NN5			29C60	24C42		44	(J4)			
(L39)	( )		45	(L34)	( )			(C15)			45	NC1			
			46					26C60	83C32		46				
(W56)			47	(W56)				(C21)			47				
			48								48				
			49								49				
			50								50				
			51								51				
C40			52	(C40)				83T22			52				
3D20	37D17		53	83D17				(T27)			53				
			54					83T17			54				
			55					(T30)			55				
(W47)			56	(W47)							56				
			57								57				
			58								58				
			59								59				
24C52	33C46		60	27C46							60	27C42			

Fig. 9 — Example of a wiring table.

this was done, each package was assigned to a specific socket position on the chassis; a certain small percentage of sockets were left unoccupied to serve as spares. All assignments were recorded on a chassis chart, which is a pictorial representation of an actual chassis (Fig. 8). The assigning of package positions was done with a view toward minimizing both the total number and length of intra-chassis wires between the sockets and the length of interchassis patch-cord.

As the next step, the available components in each package were assigned to perform the required internal functions for that particular package. The pin numbers associated with each component as it was assigned were written on the packaging diagram in such a manner that every functional signal input or signal output terminal on the package is explicitly identified as being connected to a particular pin.

A careful inventory of the components used in each and every package was kept on the chassis chart as each component was assigned. By means of this inventory the designer could know at all times throughout the process exactly how many components of every type were still available near any region of the chassis. At the end of the process of assigning components, there was usually a slight surplus of components of all types left on the chassis. If an overdraft of components occurred, however, a spare socket could be utilized to provide the needed components. Out of a total of about 21,000 diodes used in DYSEAC etched circuit packages, approximately 22% are surplus; of the 3800 delay-line segments (0.25 microsecond), about 10% are unused.

Next, the signal distribution between packages on the chassis was indicated by writing on the packaging diagram the socket number and pin number of the signal sources which were to be connected to the components that were previously assigned as the destinations of these signals. After this had been done for the complete chassis, the interchassis signal distribution was indicated. This wiring between chassis is done by patch-cords which connect to pins of special plugs that are mounted on bridges above the various chassis. A package input or output that communicates directly with some other chassis was labelled on the packaging diagram with its local bridge socket number and pin number. On the chassis chart the notation appended to the bridge socket and pin was the unique functional symbol (usually three letters) identifying the particular tube package on the packaging drawing.

The preparation of the packaging diagrams and the chassis charts was preliminary to the last step which was the ultimate goal, viz., the preparation of a set of wiring tables which enabled technicians to wire up the chassis (Fig. 9). The wiring tables consist essentially of columned pages whose entries are numbered serially corresponding to the 60 pins on a socket. Each column corresponds to a particular socket and is labelled accordingly. For example, a column headed 68A means the information entered in that column pertains to the 68th socket in chassis A. The package type is also indicated in the heading. Each item in the column pertains to a particular pin on that socket, and the entry made for each item specifies some other pin to which it is to be connected. The entries are in the form number-letter-number, specifying respectively the socket serial number, the type of signal (such as direct positive output, negative output, delayed signal), and the pin position in the socket. To help guard against errors a double entry system was employed, whereby each length of wire was represented by an entry for both of the pins that it connects. If a pin was to be connected to more than one other pin, the appropriate number of entries was made for each item.

Technicians receiving copies of the wiring tables did the wiring on all the chassis, following the wiring tables implicitly. The power wiring and other standardized wiring was done according to standard instructions appropriate to whichever of the five possible package types was specified in each column heading. The total wiring table record for the connections between the etched circuit packages of DYSEAC occupies about 270 pages (8 x 10 inches) and specifies approximately 20,000 soldered pin connections.

CONCLUSION

In the development of a large-scale computer, system design work occupies the middle ground between electronic engineering and operations analysis of the intended application. The final system design must be practicable to construct with available building blocks and current fabrication techniques, and must also be capable of satisfying the ultimate user's needs. In working out design problems concerned with reconciling these two requirements, standard methods of attack which would be generally or universally applicable probably are not obtainable. On the other hand, design problems concerned with the development of functional and construction plans, rather than with the development of system specifications, probably would provide a fruitful field for research into generally applicable procedural methods.

In particular, problems which might benefit from such research are those concerned with (1) analyzing and synthesizing the complex logical relationships existing in large-scale systems, and (2) translating such relationships into explicit wiring plans. With respect to the first class of problem, it should be noted that although methods of mathematical logic such as Boolean algebra have been found useful and effective in handling certain types of system design problems, the scope of their utility to date has been strictly limited to problems of comparatively minor importance in the system design work at this Laboratory. With respect to the second class of problem, a very promising possibility appears to lie in the idea of using digital computers themselves to carry out the processes involved in compiling wiring plans.

These detailed and tedious processes of recording, classifying, rearranging, keeping inventories, and the like, are entirely feasible tasks for present day machines.

Furthermore, in the future, utilization of computers for this purpose might not need to be limited to just the production of the wiring tables that prescribe the locations of the various pairs of soldering lugs between which connections are to be made. With the development of methods for forming wiring interconnections out of prefabricated standard interchangeable parts and with the increasing use of mechanized assembly techniques in the manufacture of electronic equipment, it is possible that the computer may one day also be utilized for the physical fabrication of the new system. That is, since the computer could produce within itself the final wiring data for the new system, it might possibly also produce mechanical patterns (e.g., punched paper cards or tape) suitable for governing the selection and positioning of the prefabricated parts out of which the inter-pin connectors might be formed. In this way, the somewhat dramatic concept of setting a computer to build a new computer might well be brought nearer to realization.

DIGITAL TECHNIQUES IN ANALOG SYSTEMS

M. A. Meyer
Laboratory for Electronics, Inc.
Boston, Massachusetts

SUMMARY — This paper discusses analog computation where the analog components consist of digital elements. Pulse rate is the quantity used to represent the data. Such computation systems may have many advantages over present analog techniques. The various components necessary to produce a complete computation system are described. Several examples of the use of these components to solve specific problems are shown.

INTRODUCTION

This paper is concerned with computers which are primarily single-purpose to solve problems of a kind that have previously been solved by analog computers. Such analog computers have been used, for example, in bombsights, navigational equipment, fire-control equipment, and in process control, to mention a few well-known applications. These computers have consisted of a combination of electronic, electromechanical, and mechanical components in varying degree depending on the specific problem. Among other things the choice of components has depended on such factors as accuracy, weight, size, speed, power dissipation, and reliability.

The more common ways of representing data in these computers have been in terms of such quantities as voltage or current, angular rotation, and angular velocity. In such computers the over-all error is an accumulation of the errors from each component. Complicated computers requiring high over-all precision have been accomplished only by the use of highly precise and expensive electromechanical and mechanical components.

The computer approach which I am now going to consider makes use of frequency as the quantity to represent the data. Such a representation has already been in use for data transmission in telemetering problems. It is also the natural output of some measuring systems, such as a doppler system. The accuracies possible in this type of computer are limited only by the conversion devices. The computing components contribute no further error if they are in operation. By proper design these components can remain operative even though there are large tolerances in the individual subcomponents. Thus the accuracy of the over-all computation can be arbitrarily set to be limited only by the conversion devices.

In many applications, too, it is possible to use inherently reliable components, such as magnetic coils, to do most of the operation. Further, it allows the possibility of reduced weight, size, power, and increased speed over present techniques.

In the systems to be discussed here, frequency, or more specifically, pulse rate, makes it possible to use digital techniques in implementing the computer. It is only the presence, or absence, of a pulse which conveys the information. The use of digital techniques is further best implemented by using a binary number system to represent the quantities. A parallel binary coding of the analog quantity can generate a pulse rate proportional to this quantity, and conversely, a pulse rate can be transformed into a parallel binary code. Once the pulse rates are available, the arithmetic operations, such as addition, subtraction, and multiplication can be easily done by well-

known pulse techniques. Integration, function generation, and inverse operations can also be performed in a straightforward manner.

DESCRIPTION OF DIGITAL COMPONENTS

The digital components necessary to accomplish all of the functions now possible in analog computers are listed as follows and then discussed more fully:

1. Conversion devices
2. Gates and buffers
3. Binary rate multipliers
4. Forward-backward registers
5. Synchronizers and timing generators
6. Diode function generating matrices.

1. Conversion devices are used to transform analog data, such as voltage or shaft position, into either a parallel binary code or a pulse frequency. Conversely, one requires converting either the pulse frequency or a parallel binary number into analog information. One should add at this point that in over-all system design the possibilities of such a computer should be considered at an early stage. In some problems it is just as easy to obtain a frequency as a measure of some quantity of interest as it is to obtain voltage or shaft position.

A dc voltage can be converted directly into frequency by one of several known techniques. One method is to make a gate whose time duration is proportional to the dc voltage by conventional linear time base techniques, and then apply this gate to the output of a pulse generator operating at a constant repetition frequency. The gate is generated periodically (the sampling rate) so that the average number of pulses per second is proportional to the voltage. If this voltage varies, one must make the sampling rate large enough to follow the variations in the voltage.

The conversion of a pulse rate proportional to a binary number into an analog voltage can be achieved by simply averaging these pulses if they are of uniform height and width, so that one has, in effect, a direct-reading frequency meter which produces a voltage proportional to the average frequency.

A parallel binary code can be converted into a voltage by means of linear addition of voltages. A *one* in a given binary place implies the addition of a voltage proportional to the binary weight of the place.

Devices for converting shaft position into a parallel binary code fall into two classes, the incremental and the continuous type. The incremental type operates on the principle of producing a pulse every time a shaft moves an amount considered to be the least significant quantity. The pulses are then counted in a register giving the binary representation of the shaft position. If the motion can be in either direction, a forward-backward register must be used.

A continuous type reads at all times the position of the shaft and is in no way affected by the possibility of dropping an increment. An example of such a converter is the coding wheel shown in Fig. 1 for a four-place binary number. Other types are available and under development which use essentially commutating segments on wheels which are geared down by factors of two from one wheel to the next. This approach is similar to a mechanical counter. The outputs of these conversion devices may be thought of as a number of switches, one for each digit represented, which are either opened or closed. The binary number represented by the state of these switches can be converted into pulse rate by the binary multiplier to be discussed later.

Fig. 1 – Four-place binary disk.

An important conversion device is the binary resolver. A binary resolver converts the sine or cosine of a shaft position to a parallel binary number. Such a device may be implemented in a number of ways. A cam arrangement which converts an angle θ into the sine or cosine of θ mechanically as a shaft position can be used in conjunction with a shaft-to-digital converter just described; or the binary output of a shaft angle from a shaft-to-digital converter can be converted into the sine or cosine of this angle by a diode function table. Another approach is to utilize a coding wheel which has been coded for the sine and cosine. One may also generate the sine and cosine by solving the differential equation for sine and cosine using θ as the independent variable.

It is possible in all these devices discussed, whether they be analog-to-digital or digital-to-analog, to obtain the inverse operation by feedback techniques. That is, once one has an analog-to-digital converter, by using feedback one can make a digital-to-analog converter.

2. Gates are used to transmit a train of pulses which appear as one input to the gate, if the other input is the binary representation of unity. Sometimes an inhibiting gate is used, which in this case the train of pulses is not transmitted if the second input is a binary one. Buffers are used to add pulses on two or more lines, when these pulses do not occur at the same time. If X_1 represents the pulse rate on a single line and X_2 represents the

ulse rate on a second line, the output on the common ne produced by buffing the two lines together represents pulse rate equal to $X_1 + X_2$. This is true if none of the ulses in X_1 are time coincident with those of X_2. Thus ne has simply effected the addition of two variables.

3. The next important element of this computer tech-ique is the binary rate multiplier. The input to a binary ate multiplier is a pulse rate and a parallel binary code. The output is a pulse rate proportional to the product of oth. Such device allows immediately for conversion from arallel binary code to a pulse rate, where in this case ne incoming pulse rate would be a constant. It also nplies the possibility of multiplying two binary numbers y successive operations on same pulse rate. In the inary multiplier, Fig. 2, one number X appears as a train f pulses on a single line. The other number Y ($O \leq Y \leq 1$) ppears as a parallel code on the n gate control lines.

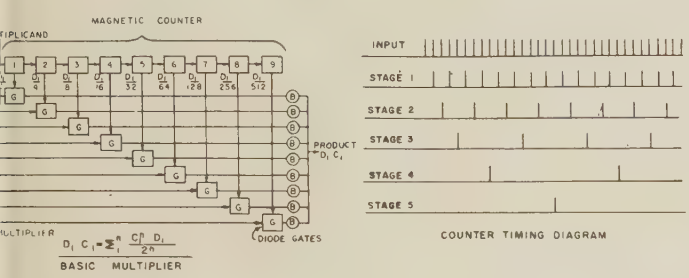


Fig. 2

The number of lines, n, determines the accuracy of the nultiplication operation. The X pulses are fed into an -stage binary counter which has as output ½ the number f pulses out of the first counter stage, ¼ the number of ulses out of the next stage, ⅛ the number of pulses out f the next, etc. The Y input, which controls the gates, letermines which counter outputsget through to be added o form the product. The product is the sum of all pulses vhich pass the gates. The multiplier thus performs the omputation,

$$\text{Product} = \sum_{n=1}^{N} \frac{X}{2^n} Y_n \qquad (1)$$

vhere the Y_n are either 0 or 1, depending on the binary ode of Y.

Addition of the pulses on the n lines of a binary nultiplier requires that no two pulses occur at the same ime at the output. In the usual cascaded binary counter rrangement the output pulse of each stage is also the nput to the next, so that the pulse timing is such that vhen a pulse appears on any one line, a pulse also ap-ears on all previous lines so that no two outputs could e added together. If the output of each stage is taken rom the side of the counter opposite to that which ad-ances the next stage, so that output pulses and ad-ancing pulses alternate in time, the timing diagram of he counter appears as shown. No two output pulses ccur at the same time. It is interesting to note that the otal number of pulses out of a binary counter is always less than the total at the input. When the multiplier is unity, so that all gates are passing pulses,

$$\text{Product} = \sum_{n=1}^{N} \frac{X}{2^n} = X\,(1 - \tfrac{1}{2}^N). \qquad (2)$$

Binary multipliers have been constructed which use a binary counter consisting of a single magnetic core, a triode normally cut off per stage, and a diode gating arrangement.

4. A forward-backward counter is a counter which counts forward for a pulse train input X on one line, and backward for a pulse input Y on a second line. The output in parallel form obtained from the stages of the counter is the parallel binary representation of the quantity $\int_0^t (X - Y)\,dt$. The pulse output, plus a polarity operator, is proportional to the difference between the two input pulse rates. Polarities in general are handled in this type of computer by having two lines, one carrying pulse rates proportional to positive quantities and the other having pulse rates proportional to negative quantities. A polarity operator associated with each operation determines to which line a pulse rate associated with the operation is connected. Forward-backward counters can be made in a conventional manner by means of a flip-flop counter in which the carries are propagated through gates in the appropriate manner depending on which input line a pulse is received from. The gates can be either diode or magnetic gates.

A typical counter stage is shown in Fig. 3. The appropriate carries are propagated according to the output of a polarity flip-flop which is set by the two input

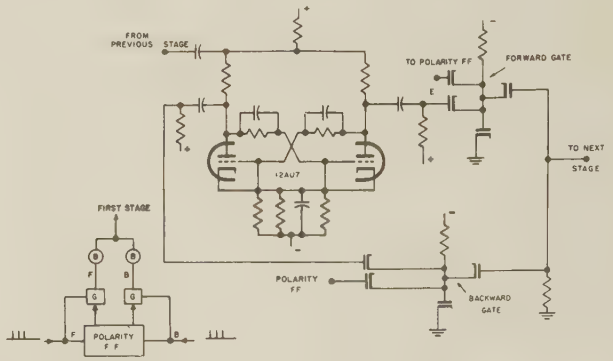


Fig. 3 — Forward-backward counter.

pulse trains. The combined output of the gates set by the polarity flip-flop is a pulse rate equal to the instantaneous difference between the two inputs and a polarity output, which tells the counter stages in which direction to count. The output pulses gated by the polarity flip-flop are fed into the first counter stage.

The flip-flop type permits static read-out and thus is useful where the contents must be continuously available, such as at the output of a computer. In many cases in

intermediate computer operations there is no requirement that the contents always be available. In the case of the binary multiplier, for example, the only time it is necessary to know whether there is a zero or one on any of the n parallel lines is when a pulse comes into, or out of, the counter. Thus, if the register has a dynamic read-out synchronized with the pulses coming into the counter, there would be sufficient information for multiplication. Such action could be achieved from a magnetic shift register, as shown in Fig. 4. Assuming the correct number to be present in the magnetic shift register, the timing would be such that when a pulse could come into the

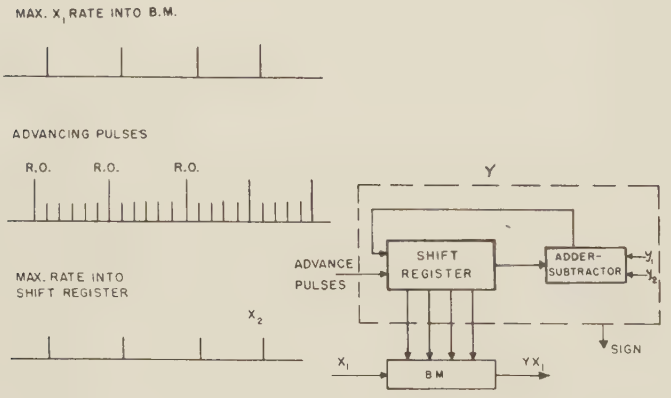


Fig. 4 – Shift register multiplier.

counter, the register would receive an advancing pulse which would read pulses out in parallel form representative of the code in the register. The input to the register would be so timed that if there were a pulse present at the input, representing one digit, it would be added or subtracted in a serial adder to the number now being read out of the register by successive advancing pulses. This number is then read back into the register and is in the register by the time the next pulse could enter the counter. Note here that the pulse handling rate of the counter must be approximately n times faster than the basic pulse frequency, where n is the number of binary places. In some applications, where the variables change at low rates, such a system is possible and desirable since magnetic components can be used to a large degree.

5. A timing generator generates a multiplicity of pulse trains, all at the same frequency, but so timed that none of the pulses of one train are time-coincident with those of another train. One train is generally associated with a particular variable or channel.

6. If the data pulses occurring on a given channel are random, or if their time of occurrence is not specified, they may be shifted in time by a synchronizer. A synchronizer has as output the same average pulse rate as the input, but the pulses occur at specified times. The inputs to a synchronizer consist of data pulses and timing pulses at a rate higher than the highest possible data rate. Logically a synchronizer is a memory device in which a data pulse is always read in regardless of its time occurrence. A timing pulse, since it will always occur before the next data pulse, will read it out.

A simplified schematic diagram of a synchronizer for one data channel is shown in Fig. 5. One dual triode and a single-core bistable element are used. Pulses to be shifted (data pulses) appear at the grid of V-2, pulsing this tube and causing current to flow in L-3. This current changes the bistable magnetic core to its other stable state and produces a negative pulse at the output winding (subsequent circuits must be arranged so as to be unaffected by negative pulses) and at L-2. The induced

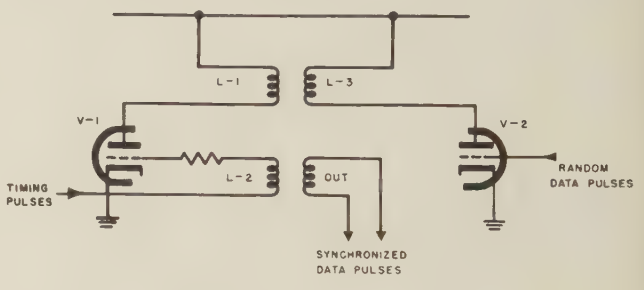


Fig. 5 – Schematic diagram of single-tube synchronizer.

negative pulses at L-2 are larger than the incoming positive timing pulses which must pass through L-2 in series to get to the grid of V-1. Thus a timing pulse cannot cause V-1 to conduct during a data pulse, and ambiguous counts are eliminated. Since the timing pulses occur at a higher repetition rate than any data pulses, another timing pulse will occur next and pulse V-1, causing the core to revert to its previous state and read out a positive pulse at the output winding. Subsequent timing pulses will have no effect on the circuit until after the next data pulse has again changed the state of the core.

7. A function table, or matrix, is made up mainly or entirely of diodes and resistors. Instead of delivering a single output when certain input conditions are satisfied, it delivers a code output on a number of lines, a different code for each binary code on the input lines. Function tables may be used for squaring numbers, multiplying by constants, generating sine or cosine, etc. The output number, of course, cannot have a greater accuracy than the input number which determines it.

APPLICATIONS TO PARTICULAR PROBLEMS

Now that the various components which are used to implement the various computer functions have been discussed, it is well to describe several arrangements using these components which can perform desirable functions. Fig. 6 shows a representation of the various components discussed.

It has been stated that a pulse rate can be transformed into a parallel binary number. Such a transforma-

on can be effected by using a forward-backward register d a binary multiplier (Fig. 7). One input to the forward-ckward register is the pulse rate NX_1, which is proportional to the binary number N. The input to the binary ltiplier is the reference pulse rate X_1, and the contents of the forward-backward register. The forward-backrd register is fed at its second input with the output of e binary multiplier. The register is then servoed until e input rates of each input are the same. This implies t the number N is then in the register.

Division is performed in a similar manner. If it is sired to divide X by Y, the equation $ZY - X = 0$ is

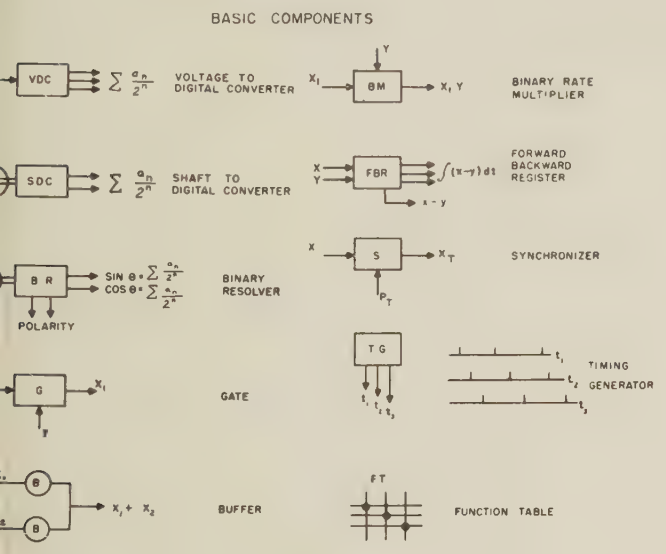


Fig. 6

olved. The contents of Z of the register are multiplied Y in successive binary multiplications, producing a ulse rate proportional to ZY, which is fed back into one put of the forward-backward register. The other input a pulse rate proportional to X, so that when the register aches a steady state the equation is solved.

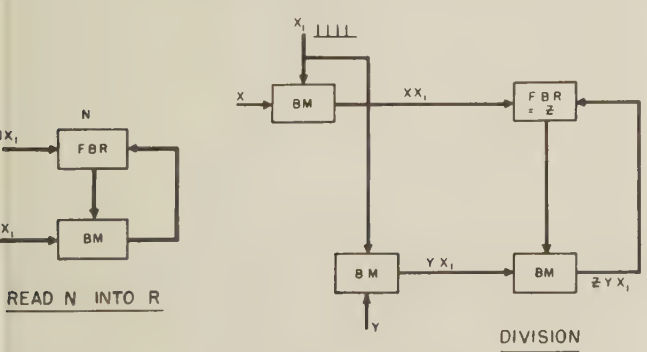


Fig. 7 – Implicit operations.

The process of division is one example of implicit nction generation, where an explicit function is derived om the corresponding implicit function. Such a technique as had considerable use in analogue computers, and may e used in these computers for generation of roots of umbers and other functions which are solutions of fferential equations.

I will now discuss the applications of these techniques to certain type of problems. The first problem is the solution of the differential equation for the azimuth angle θ of an aircraft with respect to north as seen from a moving ship.[1]

This equation

$$\frac{d\theta}{dt} = \frac{1}{R}\left(V_a \sin\left[\phi - \theta\right] + W \sin\left[\tau - \theta\right]\right) \qquad (3)$$

has been solved in certain systems using conventional analog techniques. Here ϕ is the heading angle of an aircraft with respect to north, V_a is the horizontal true air speed, τ is the direction of wind with respect to north, W is the wind speed, and R is the radar range. All quantities are available as shaft positions. The solution as shown proceeds as follows (Fig. 8): The angle $(\phi - \theta)$ is fed into the shaft of a binary resolver producing a binary coded output proportional to sin $(\phi - \theta)$, and a polarity output giving sign information. Similarly, out of a

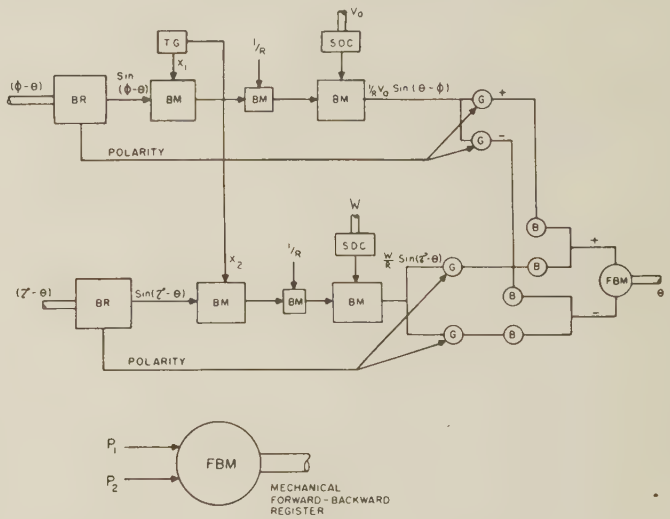


Fig. 8 – Differential equation solution.

second binary resolver is produced the binary code for sin $(\tau - \theta)$ and its associated sign. The outputs of the binary resolvers serve as inputs to binary multipliers #1 and #2. A pulse train X_1 is fed into the binary multiplier #1 and the pulse train X_2, which is displaced in time from X_1, is fed into binary multiplier #2. Both X_1 and X_2 represent the same pulse rate and are produced by the timing generator. The outputs of each binary multiplier are fed into a second set of binary multipliers with the input $1/R$ obtained in binary form by a shaft digital converter, so that the multiplication by $1/R$ is accomplished. The new outputs are then to be multiplied by V_a in the case of the upper channel, and W in the case of the lower channel. Since both V_a and W are available as shaft positions, a shaft-to-digital converter is used for

[1] Radiation Laboratory Series, vol. 21, p. 190.

each and serves as the input to a third set of binary multipliers. The two pulse rates are also fed into the third set of multipliers so that at the output of channel 1 is a pulse rate proportional to $V_a/R \sin(\phi - \theta)$, and the output of channel 2 is a pulse rate $W/R \sin(\tau - \theta)$. The polarity operators from the binary resolvers then serve to switch the binary multiplier outputs into the proper line depending on the polarity of the associated sine. The two positive pulse rates are then added together as well as the two negative pulse rates. The combined outputs are then fed into a forward-backward mechanical register. A forward-backward mechanical register consists of a shaft that is driven in incremental fashion by the momentary closing of a relay. Two relays are used; one for

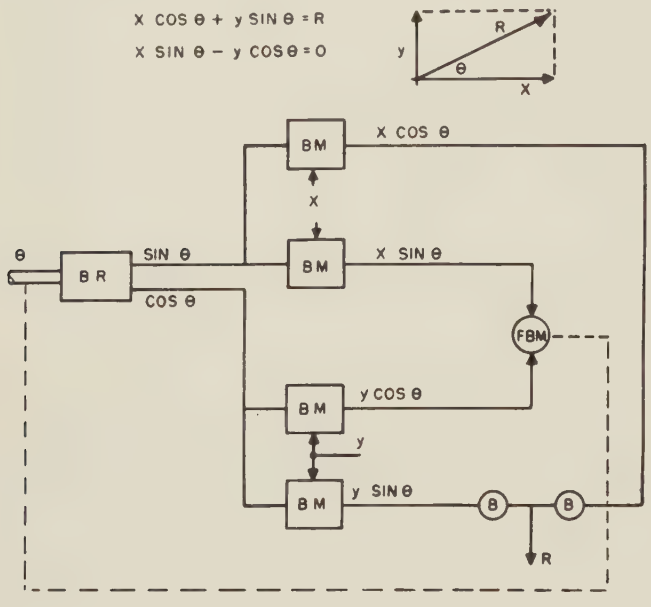


Fig. 9 – Vector solution.

producing forward increments, and the other for producing backward increments. A commercially available relay drive of this nature is the Ledex drive. The shaft output is then positioned as a result of this integration to the angle θ. It might be noted that the mechanical binary resolver could have been eliminated, as well as the mechanical forward-backward register. If an electronic forward-backward register were used to represent the angle θ and the resolution done by means of a diode function table, in this case the shaft-to-digital converters would be the only mechanical elements in the system.

The system just described is a closed-loop system. An open-loop system performing essentially the same computation would be designed approximately the same if the angle θ were a given angle and it were desired to obtain an output $\propto$ which was the integral of

$$\propto = \int_o^t \frac{1}{R} \left[V_a \sin(\phi - \theta) + W \sin(\tau - \theta) \right] dt \quad (4)$$

Open-loop systems essentially process the data as it arrives without using storage for auxiliary computations. For this reason the techniques have been called "operational digital."

A second example of a closed-loop system is that of a right triangle where it is desired to find the hypotenuse and angle of the triangle (Fig. 9). Given the two variables X and Y in pulse rate form representing the two legs of the triangle, the equation $X \sin\theta - Y \cos\theta = 0$ must be solved. A binary resolver generates the $\sin\theta$ and $\cos\theta$, which are fed as inputs to binary multipliers. X and Y are also fed into the binary multipliers, so that the output of a binary multiplier producing $X \sin\theta$ is fed into one input of a forward-backward mechanical register, and the sec-

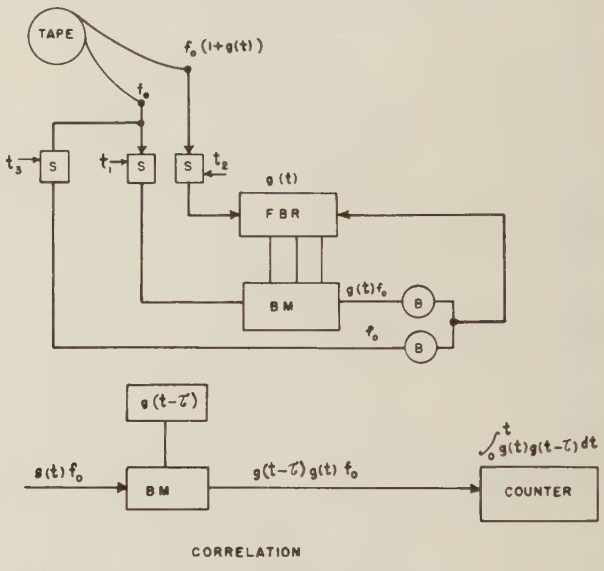


Fig. 10

ond input is the pulse rate representing $Y \cos\theta$. The output of the forward-backward mechanical register is coupled to the binary resolver so that the system servoes itself to the angle θ, which solves the equation. The rates representing $X \cos\theta$ and $Y \sin\theta$ are then added together to produce a rate which represents the hypotenuse of the triangle.

Another problem might be the correlation of information that is recorded on tape in frequency modulation form (Fig. 10). The quantity of interest is represented as $G(t)$, assumed always positive for this example, so that the frequency appearing on the tape is $F_o\,[1 + G(t)]$. F_o is also recorded. By means of a forward-backward register and a binary multiplier, a servo system can be set up as shown so that the contents of a forward-backward register are equal to $G(t)$. The output of the binary multiplier is $G(t)F_o$. $G(t-\tau)$ can be obtained in a similar manner by using the delayed information from the tape. $G(t)F_o$ is then fed as one input into a binary multiplier and $G(t-\tau)$ is the second input producing a pulse rate proportional to the product of $G(t)$ and $G(t-\tau)$. These pulses are then fed into a counter producing the desired integration.

A final example is that of a digital servo where it is sired to obtain a shaft position proportional to some ıction $F(t)$, (Fig. 11). $F(t)$ is represented in a parallel ıary form and serves as one input to a binary multiplier l by a pulse rate X_1. The desired shaft position is

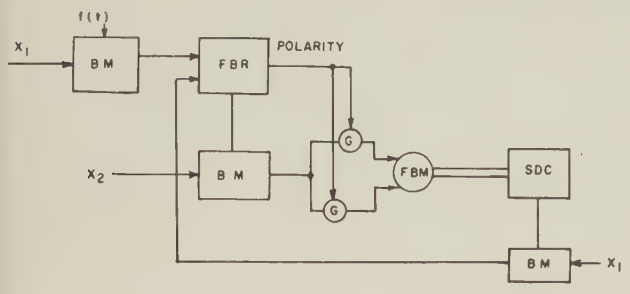


Fig. 11 – Digital servo.

upled to a shaft-to-digital converter whose output feeds binary multiplier which has as second input a pulse te X_1. The outputs of the two multipliers are fed into a rward-backward register which might be called the ve- city register for the system. The contents of this ed into a second binary multiplier fed by a pulse rate . The output of this binary multiplier goes to a ± bus pending on the sign of the contents in the forward-back- rd register. These two outputs are then fed into a rward-backward mechanical register. It can be seen that the function $F(t)$ is moving at constant velocity, the ntents of the velocity register will be a fixed number d the output shaft will be driven at the proper rate having essentially no velocity or position error. Some damping must be provided for loop stability and this is usually obtained by also providing a direct pulse path to the position register.

CONCLUSION

I have described the application of digital techniques to systems which have been previously considered as analog systems. It is felt that the use of these techniques can lead to a higher order of reliability and ease of manufacture, with a considerable reduction of weight and power over present day systems. Most of the components are readily adapted to the use of transistors when these are available in reliable form. There is still considerable room for the development of small, simple conversion devices, but this development will proceed as the application of this technique becomes more extensive. It is to be emphasized again that in the design of systems where such a technique would be useful, many conversion devices might be eliminated by proper representation at the source of the measurement.

ACKNOWLEDGMENT

I wish to acknowledge the work of Bernard M. Gordon and Renato Nicola for the contributions made to this technique for arriving at practical realizations for the components, as well as for contributing many ideas for "operational digital" implementation.

A HIGH-SPEED CORRELATOR*

Harold Bell, Jr.** and V. C. Rideout
University of Wisconsin
Madison, Wisconsin

SUMMARY – The correlation function,

$$\phi_{ij}(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_0^T f_i(t) f_j(t-\tau)\,dt$$

is of great interest today because of its use in the fields of oceanography and meteorology and because of its recent applications in the field of communication. Various machines, both analog and digital, have been designed for the automatic computation of correlation functions. The machine described in this paper differs from those which have previously been described in the literature in that the speed with which it computes the integral above for each value of τ_i is mainly limited by the minimum value of T permissible for a precision of a few per cent.

The high-speed correlator is of the analog type and utilizes the following main elements:

1. A high-speed (or wide-band) multiplier.
2. A high-speed integrator.
3. A high-quality lumped-constant delay line with a total delay of 2080 microseconds.
4. A switching scheme which permits the delay τ_i to be varied rapidly in discrete steps.

With these units the correlator developed allows a correlation function of 41 discrete points to be computed for functions occupying nearly the complete audio spectrum, in a period of only five seconds.

A number of tests of this machine operating as an auto-correlator and as a cross-correlator have been made. Theoretical studies regarding precision have been checked by such tests for simple functions, and show that a precision of 2% is possible. Separation of a sine-wave signal from noise was found possible for S/N ratios of as low as -25 db.

In addition to other possible uses, the high speed of this correlator has raised the possibility of using it to continuously process data in cases where, for example, a signal is masked by noise. With the aid of two tape recorders a single high-speed correlator should make it possible to continuously process the data in a band 360 cycles wide, with a two second delay. Signals occupying wider bands would require more correlators for continuous processing, but could be handled if they are known to occupy the band for only certain calculable percentages of the time.

I. INTRODUCTION

The importance of the correlation functions in communication and measurement is well recognized today.[1,2] A number of machines for the automatic calculation of these functions have been described.[2,3,4,5,6] These machines are of two main classes – digital and analog. In general the digital correlators which have been described in the literature are quite slow,[6] and the analog correlators, though somewhat faster, leave much to be desired.

The analog type of correlator to be described in this paper was designed to operate at as fast a rate as possible. It was found after construction of the machine that the rate achieved (41 points in 5 seconds) was close to the theoretical maximum speed possible if errors were to be kept in the vicinity of from one to five per cent.

II. BASIC REQUIREMENTS OF AN ANALOG CORRELATOR

The simplest correlation functions have the form

$$\phi_{ij}(\tau) = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} f_i(t)\, f_j(t+\tau)\,dt, \qquad (1)$$

where $i = j$ for the auto-correlation function, $i \neq j$ for the cross-correlation function, and $f_{i,j}(t)$ are stationary random processes. In the case of real functions it may be shown that $\phi_{ij}(\tau)$ is symmetrical about the origin and that (1) may be replaced by

$$\phi_{ij}(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_{0}^{T} f_i(t)\, f_j(t-\tau)\,dt. \qquad (2)$$

It is not necessary in practical cases to integrate for a very long period of time. Thus if $f_i(t)$ contains sinusoidal components of maximum period T_1, it is only necessary to integrate for a time equal to T_1,

$$\phi_{ij}(\tau) = \frac{1}{T_1} \int_{0}^{T_1} f_i(t)\, f_j(t-\tau)\,dt \qquad (3)$$

* This paper includes material from a thesis by Harold Bell, Jr., presented in partial fulfillment of the requirements for the Ph.D. degree at the University of Wisconsin.

** North American Aviation Inc., Downey, Calif.

[1] N. Wiener, *"Extrapolation, Interpolation and Smoothing of Stationary Time Series,"* John Wiley and Sons, Inc., New York; 1950.

[2] Y. W. Lee and J. B. Wiesner, "Correlation functions and communication applications," Electronics, vol. 23, p. 86; 1950.

[3] M. J. Tucker, "A Photoelectric correlation meter," *Jour. Sci. Instr.*, vol. 29, 1952.

[4] V. J. Guethlen and E. W. Hix, "Correlator for Low Frequencies," Goodyear Aircraft Corp., Aerophysics, Dept., Akron, Ohio.

[5] H. E. Singleton, "A Digital electronic correlator," Tech. Repts. MIT Electronics Res. Lab., no. 152.

[6] M. J. Levin and J. B. Reintjes, "A Five-channel electronic analog correlator," Proc. Nat. Elec. Conf., vol. 8, p. 647; 1952.

We will assume that the functions $f_{i,j}(t)$ are in continuous rather than discrete (or digital) form. An analog machine can compute the correlation functions for discrete values τ_i of the variable τ by the following processes.

1. Delay $f_j(t)$ successively by amounts τ_i, where $\tau_i = k \triangle \tau$, $k = 0, 1, 2 \ldots, n$.
2. For each value of τ_i multiply $f_i(t)$ by $f_j(t-\tau)$ for a time T_1.
3. For each value of τ_i integrate this product for a time T_1, and divide the result by T_1 to obtain $\phi_{ij}(\tau_i)$.

Thus the analog components required are, as shown for the auto-correlator case in the simplified block diagram of Fig. 1:

1. a multiplier,
2. an integrator,
3. a scale-changing amplifier to multiply by the constant $1/T_1$,
4. a delay line whose delay can be varied in steps.

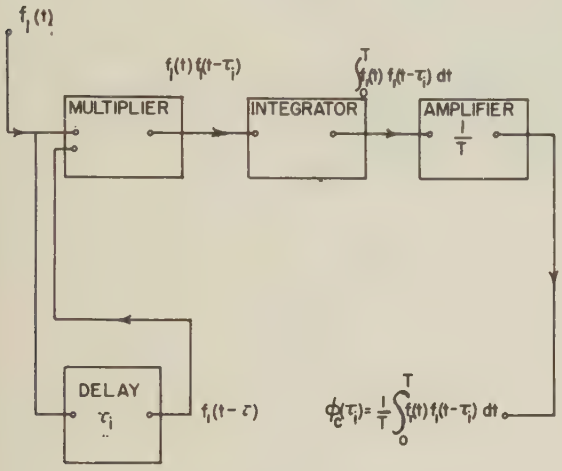


Fig. 1 — Block diagram of an auto-correlator.

For high speed operation it is desirable that the multiplication, integration and scale-changing be performed simultaneously and rapidly for each value of τ. Philbrick computer units[7] were therefore chosen for the first three components listed above. These components were the M-unit, a quarter-square multiplier[8]; the J_C-unit, a clamped integrator; and the C-unit, a scale changer.

The remaining item, a delay line, had to be designed and built because of its rather unusual requirements.

III. DELAY-LINE DESIGN

The unit of time used in the integrator mentioned above is 1/2400 sec, hereinafter referred to as 1 psec. It was decided to build an artificial delay line with 40 steps and a total delay of 5 psec. Preliminary calculations showed that the resultant delay-line bandwidth would be of the order of 15 kc, which is roughly the useful bandwidth of the high-speed computer components described above.

The delay line design decided upon was one which used 40 m-derived T-sections with m = 1.49 together with 40 BT (or bridged-T) phase-compensating sections, according to a design described by Turner.[9] It was found however that in order to obtain proper matching the two types of sections had to be segregated, and an m-derived matching half-section used between the T-sections, which have a characteristic impedance which varies with frequency, and the BT sections, which are all-pass networks of constant characteristic impedance. Another matching half-section was required to obtain a constant input impedance at the input of the T-section chain, as shown in Fig. 2. This figure also shows the 41 switches required to successively add T-sections and BT-sections in pairs and to give the 40 steps of delay.

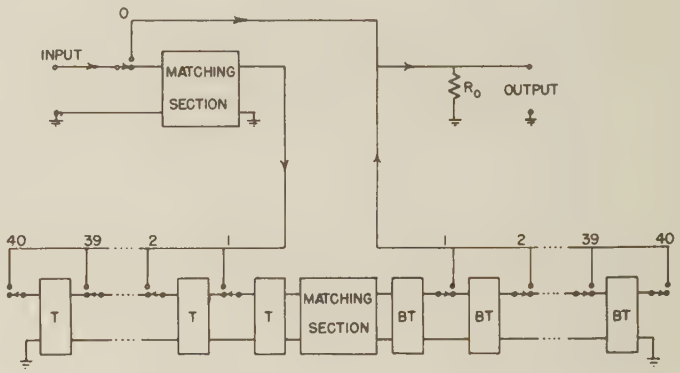


Fig. 2 — Block diagram of the delay line.

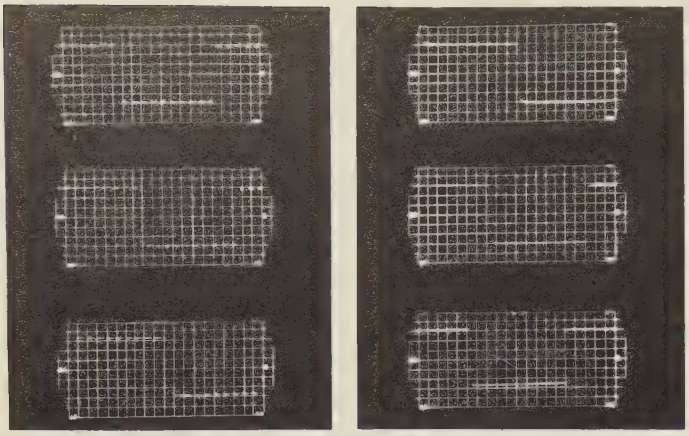
The delay line was built using high-Q toroidal coils with powdered iron cores, and had a characteristic impedance of 524.5 ohms and a cut-off frequency of 18.25 kc. The phase was theoretically flat to 0.82 of cut-off, or 15 kc. Square wave tests (Fig. 3) showed that the line performed well.

The desired operation of the correlator required that the steps in delay be set in rapidly. A stepping switch of the kind used in telephone work was chosen to successively move switches 0 through 40 (Fig. 2) to their "up" positions. This stepping switch is driven by a square wave derived from a low-frequency oscillator. A clamping signal obtained from this square-wave is used as shown in Fig. 4 to clamp the integrator output to zero for a time T_0, and then unclamp it for a time T_1 during the period $T = T_0 + T_1$ on each step. The minimum time T_0 was selected as 10 psec, or twice the line delay, to permit reflected waves on the line caused by switching to die out before computation begins at each step. For

[7] Catalog and Manual, Geo. A. Philbrick Researches, Inc., Boston, Mass.; 1951.

[8] H. Bell and V. C. Rideout, "Precision in high-speed electronic differential analyzers," Proc. Cyclone Symp. II, Reeves Inst. Corp., New York; 1952.

[9] A. H. Turner, "Artificial lines for video Distribution and delay", RCA Rev., vol. 10, p. 477; 1949.



(a). PHILBRICK SQUARE WAVE WITH MARKERS
UPPER: ZERO DELAY
MIDDLE: 2.5 p SEC. DELAY
LOWER: 5 p SEC. DELAY

(b). PHILBRICK SQUARE WAVE
UPPER: 5 p SEC. DELAY
MIDDLE: 2.5 p SEC. DELAY
LOWER: ZERO DELAY

Fig. 3 – Square wave tests of delay line.

example, the time T_1 was ($\frac{1}{8}$ - $^{10}/_{2400}$) sec, for an 8-cycle-per-second drive, or 290 psec. It was found that this time could be reduced without much sacrifice in precision, as will be shown in the next section.

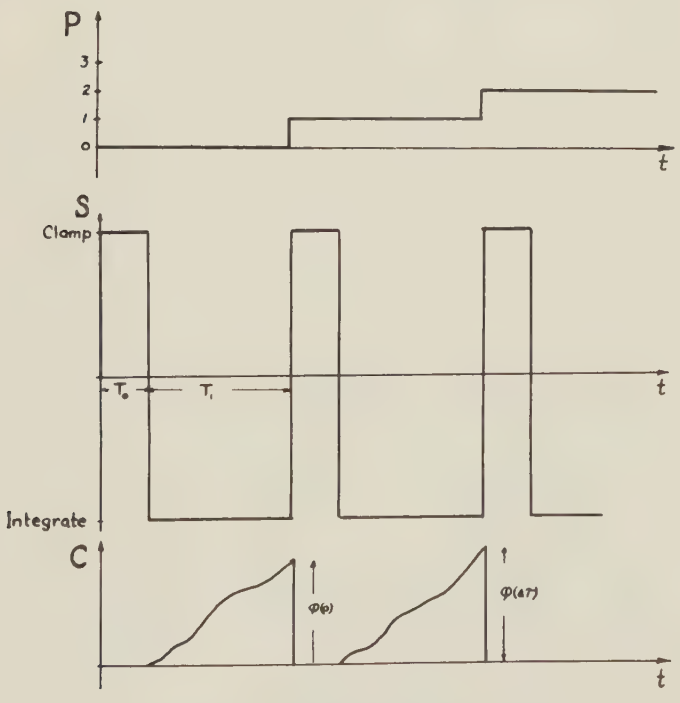


Fig. 4 – (a) Stepping Switch position, P.
(b) Clamping signal, S.
(c) Correlator output, C.

The delay line has a total attenuation of 21.4 db, and this had to be compensated for. A second set of contacts on the stepping switch was used to make a logarithmic attenuator which added 21.4 db of attentuation on the zero step, ($^{39}/_{40}$) x 21.4 db on the first, etc. Still another set of contacts was used to provide a signal which increased step-wise with τ_i to provide the sweep voltage on the cathode ray oscilloscope used as an output device.

Summarizing the operation described above,

$$t = T_0 + T_1 = T \text{ in Fig. 4,}$$

the stepping switch has just moved from position 0 to position 1, introducing delay $\Delta\tau$, removing $^1/_{40}$ of the decibel attenuation used to compensate the line attenuation, and moving the cathode ray trace $^1/_{40}$ of the total desired distance on the screen. At $t = 2T_0 + T_1$ the integrator is unclamped, and the correlation integration proceeds for a time T_1, at which time the integrator output is again clamped to zero.

The complete correlator, in block diagram form, is as shown in Fig. 5.

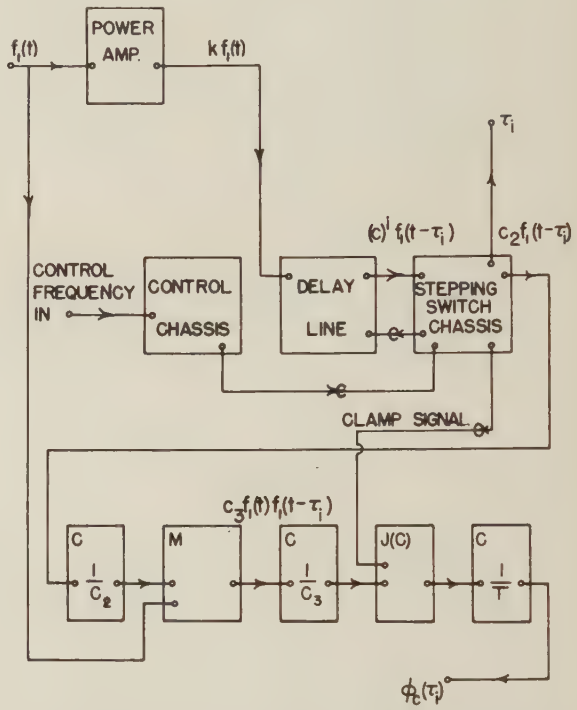


Fig. 5 – Block diagram of the high speed correlator. Connections are shown for the auto-correlation case. To compute cross-correlations connect $f_2(t)$ to the multiplier M.

IV. ERROR ANALYSIS

It is obvious that considerable error will result if an attempt is made to find correlation functions of a signal which contains frequencies greater than about 15 kc, because the line delay will sharply increase for frequencies any closer to its cut-off frequency. At high frequencies there will also be phase shift errors in the other computer components.

If the frequencies of interest all lie below 15 kc the error due to the finite time of integration T_1 is of relatively more importance. Davenport[10] has obtained an expression relating the integration time to the ratio of the

[10] W. B. Davenport, "Correlator errors due to finite observation intervals," Tech. Repts. MIT Electronics Res. Lab, no. 191.

verage value of the output of a correlator to the rms alue of its error. This expression however requires that ie auto-correlation function of the signal be known. It as therefore decided to experimentally observe errors in ie correlator output for the case of auto-correlation of hite noise for a fixed value of τ_i as T_1 was varied. The robable percentage error based on a Gaussian distribuon of error was determined by making 82 runs for fixed alues of τ_i ($\tau_i = 0$, $\tau_i = 2.5$ psec) for each of several alues of T_1 as shown in Fig. 6. Note that the probable rror decreases sharply at first with increase in T_1, but iat further decrease is slow after an error of 1% is eached at about 106 psec.

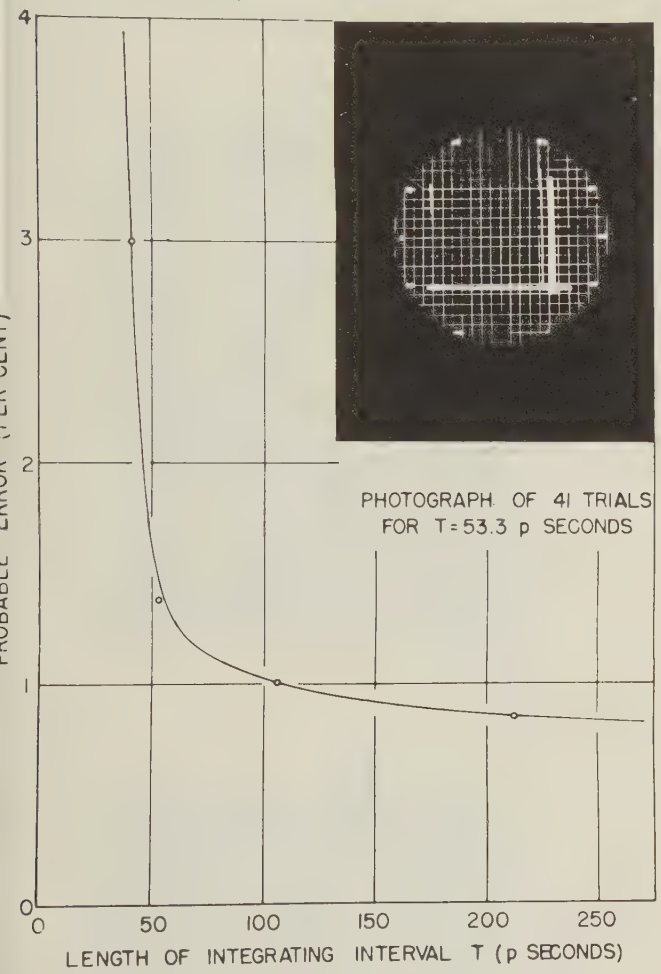


Fig. 6 — Probable correlator error due to finite integration interval versus length of integrating interval.

In the case of auto-correlation of a sinusoid the rror due to finite time of integration may be calculated,[11]

$$|\epsilon_1| \leq \frac{3}{4T_1} \tag{4}$$

This error is therefore ¾ percent for $T_1 = 100$ psec, nd this is less than the error in the rather general case f auto-correlation of white noise. Thus it was assumed hat for any input functions the error due to finite time of ntegration might be expected to be of the order of the usual figure of 1 to 2 per cent found in most electronic analog equipment.

It is not possible in any case to increase the time of integration T_1 at will because of an inherent defect in the operational integrator. The gain K of the d-c amplifier used in the integrator is limited to about 15000 by the wide band required for high speed operation. For a sine wave input the absolute integrator error varies directly with the integrating interval T_1 and inversely with the amplifier gain K,

$$|\epsilon_2| \leq \frac{T_1}{4K} \tag{5}$$

This error,[11] plus the calculated error for the sinusoidal case due to the limitation in T_1 as given by (4) results in the overall curves plotted for three values of K in Fig. 7. Note that for $K = 15000$ a minimum value of error is obtained at $T_1 \approx 200$ psec.

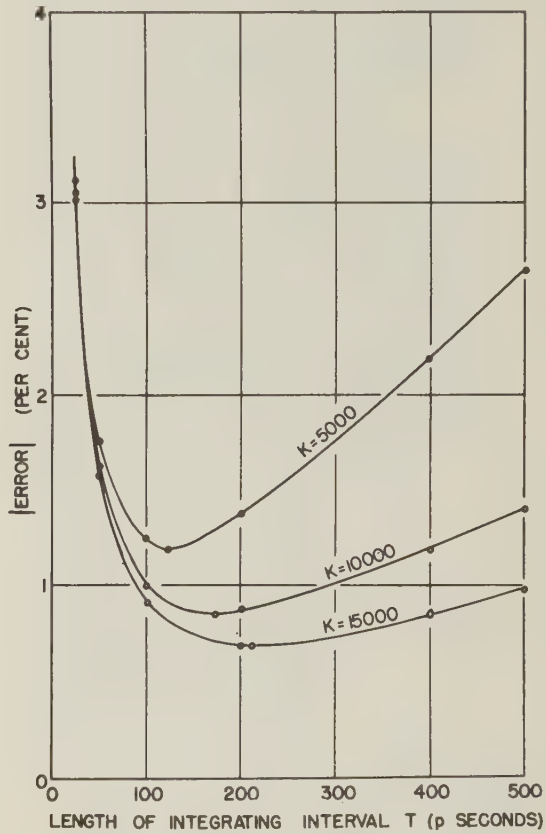


Fig. 7 — Calculated error (for auto-correlation of a sinusoid) due to finite interval of integration and finite gain K of the operational amplifier integrator for three values of K.

In practice one of the largest sources of error lay in the multiplier, which was difficult to align and which tended to drift out of proper adjustment. It is hoped that some new high speed multipliers now under development[12] and the use of automatic zero stabilization on all amplifiers will result in reduction of these errors.

[11] H. Bell, "The Design and some applications of a high-speed correlator," Ph.D. Thesis, University of Wisconsin; 1953.

[12] K. L. Hendrickson, "Thyrite Squaring Circuits for High-Speed Analog Computer Use," M.S. Thesis, University of Wisconsin; 1953.

V. EXPERIMENTAL RESULTS OBTAINED WITH THE HIGH-SPEED CORRELATOR

Sine-waves and square waves were used to check the operation of the correlator. If the input is of the form

$$f(t) = B_n \cos (k_n t + A) \tag{6}$$

then by use of (2) the auto-correlation function may be shown to be

$$\varphi_{11}(\tau) = \frac{B_n^2}{2} \cos k_n \tau \tag{7}$$

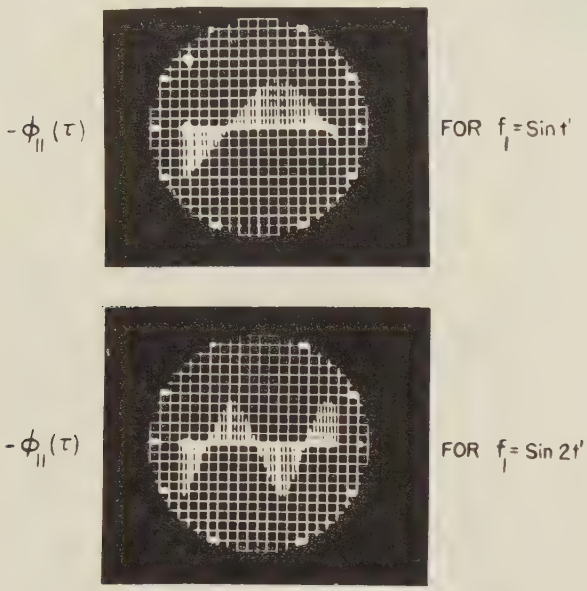


Fig. 8 – Photographs of correlator output for sinusoidal input.

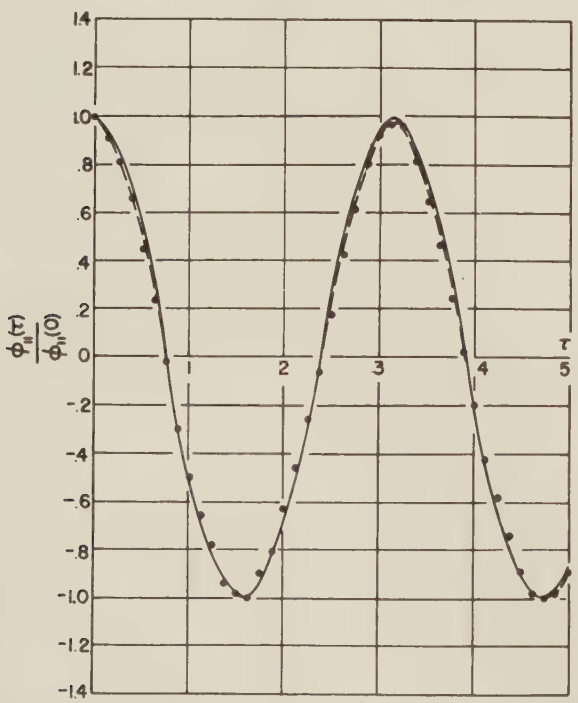


Fig. 9 – Experimental correlator output points (from Fig. 8) versus calculated $\phi_{11}(\tau)$, (Solid curve) for sinusoidal input.

Results for two cases are shown in Fig. 8, and the first of these cases is compared with the calculated value of $\phi_{11}(\tau)$ (solid curve) in Fig. 9. The auto-correlation of a rectangular wave with unequal positive and negative parts is shown in Fig. 10. Results are compared with calculated values in Fig. 11. The scattering in the second case in these figures for $\tau > 4.3$ was caused by a temporary difficulty with contact resistance in the stepping switch wafer.

Separation of sine-waves from noise by auto-correlation was also examined, with the results shown in Fig. 12. Here the sinusoidal signal was introduced halfway through the correlator cycle. Noise was held at fixed amplitude and the signal successively reduced in ampli-

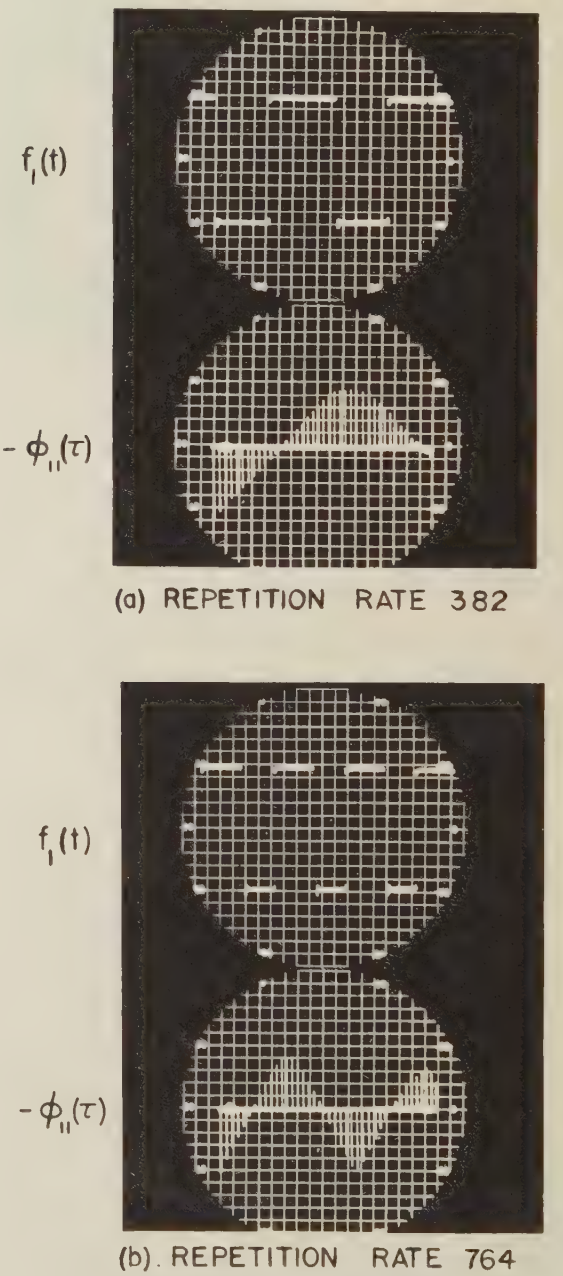
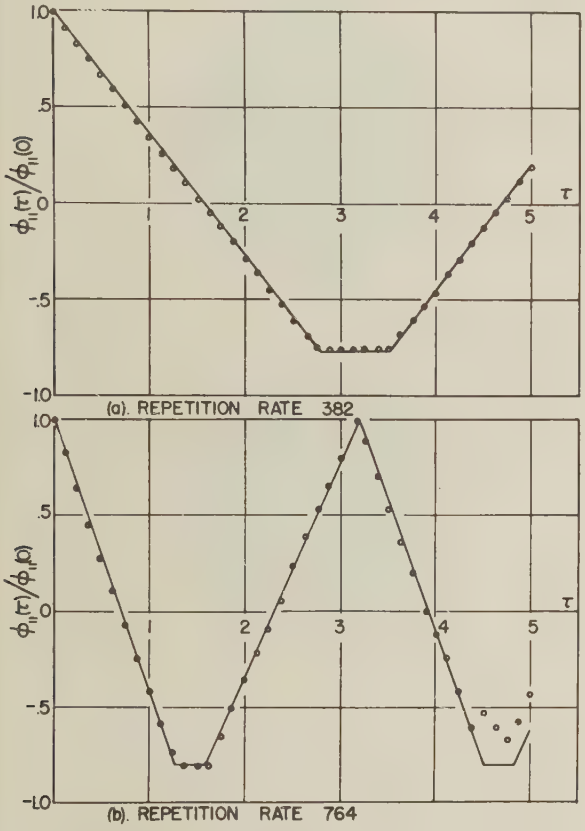


Fig. 10 – Photographs of square-wave input and corresponding output of the correlator.

le in these tests. Results appear to be in rough agreeent with theoretical results obtained by Lee, Cheatham d Weisner.[13]

The time response of a linear system to a unit imlse may be obtained by cross-correlating the input and tput to the network where white noise is used for the ut signal. The same results for the case where the twork was a simple R-C circuit are shown in Fig. 13, three values of the time-constant RC. The impulse sponse of an R-C network is of the form $K\epsilon^{-t/RC}$ d it may be seen that the experimental results have s form.



g. 11 — Experimental correlator output points (from Fig. 10) versus calculated $\phi_{11}(\tau)$, (solid curve) for square-wave input.

VI. CONTINUOUS PROCESSING OF SIGNALS WITH THE HIGH-SPEED CORRELATOR

From Sec. IV it may be seen that integration for a riod of say 100 psec or 41.7 milliseconds is adequate, precision of a few percent, but that any appreciable rection in this figure will lead to an undesirable amount error. Another 10 psec must be provided for the calcuion of each point in the correlation function as pointed t in Sec. IV. Thus a total minimum time of 4510 psec approximately 2 seconds is required to determine the points (including $\tau_i = 0$) possible with this machine.

Actually the machine required 5 seconds to properly compute the 41 points, faster operation being undesirable because of chatter in the relays, which were not specially chosen for high speed operation.

This high speed of operation makes it possible to consider the application of this correlator to the continuous processing of signals. Such operation may be desirable where it is desired to discover repetitive signals masked by noise, as in radar and sonar, or to discover hidden periodicities in meteorological data, such as ocean waves. It is necessary to make 41.7 milliseconds of signal repetitively available to the correlator for the

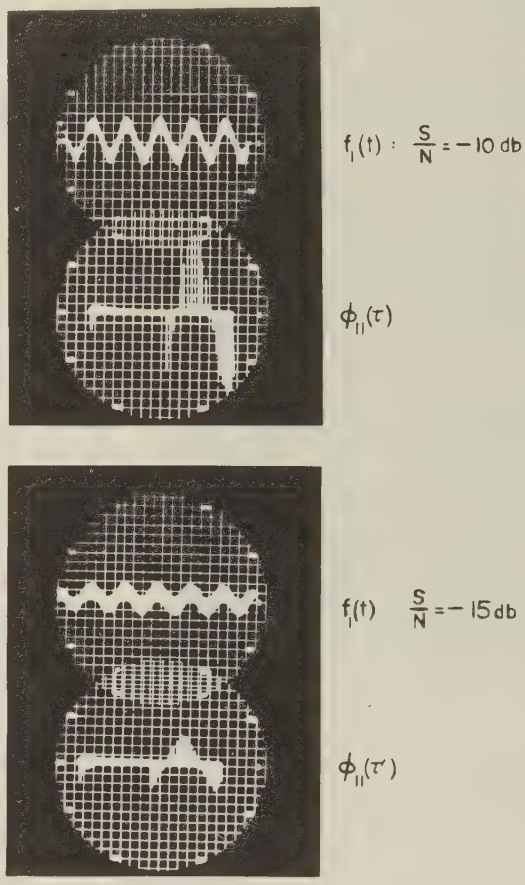


Fig. 12(a) — Photographs of the input and output of a correlator for auto-correlation of a sinusoid plus noise (noise peaks are not visible in photographs).

2-second period required to process the signal. Thus a recording might be made on tape for 2 seconds and the tape might then be speeded up to reduce its duration to 41.7 milliseconds, a speed-up factor of about 48 times. It could then be fed repetitively into the correlator for 2 seconds while another tape recorded for 2 seconds. The allowable bandwidth would have an upper limit 48 times smaller than the highest frequency which the line could handle, or about 313 cycles. The lowest frequency would be 48 times smaller than the frequency at which the line is one wavelength long, or 10 cycles. (This could be reduced by using a line with a greater number of sections.)

Signals with components occupying bandwidths greater than about 300 cycles could be processed by one

Lee, Cheatham and Wiesner, "Application of correlation analysis to the detection of periodic signals in noise," Proc. I.R.E. vol. 38, p. 1165; 1950.

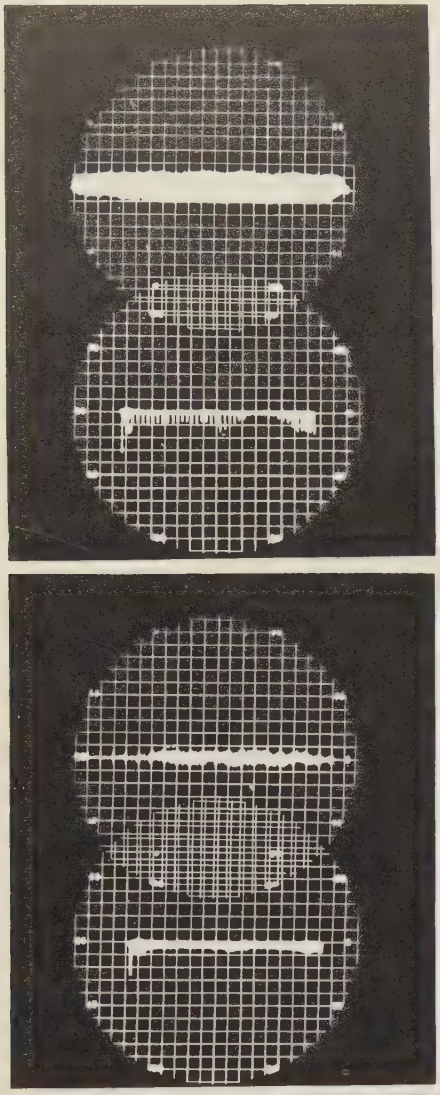


Fig. 12(b) – Photographs of the input and output of a correlator for auto-correlation of a sinusoid plus noise (noise peaks are not visible in photographs).

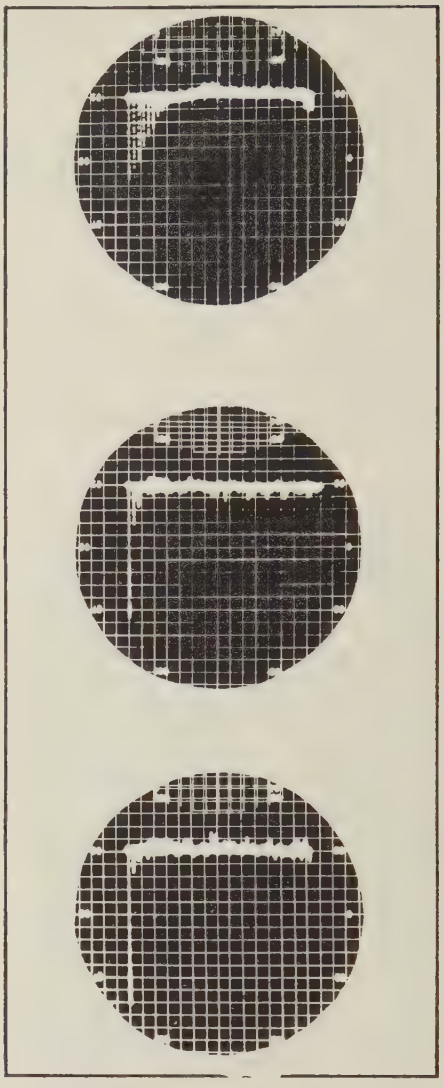


Fig. 13 – Photographs of the cross-correlation between input and output of an rc circuit with white noise input.

correlator at a slower rate than the information was received, or a number of identical correlators might be used.

It is interesting to compare the rate of transmission of information in a system with ideal coding with the rate possible where the correlator is used with binary pulses. In the ideal coding case, according to Shannon,[14]

$$H = BT \log_2 (1 + S/N) \text{ bits} \tag{8}$$

where H is the information in bits transmitted in time T in a channel of bandwidth B in which the signal to noise ratio is S/N.

A single correlator will continuously process a bandwidth of 303 cycles, determining 41 points in 2 seconds as pointed out above. If the S/N ratio at the correlator output is such that only binary digits can be discerned, then $H = 41$ bits for $T = 2$ sec. The equation above indicates that the ideal coding system will permit 41 bits of information to be handled in 2 seconds in a channel of width 300 cycles with S/N as low as -13.3 db. If only one bit of information were to be determined in this period the possible S/N ratio in the ideal coding system would be -29.4 db. In the S/N studies made with this correlator the presence of a single sinusoid, corresponding to a single bit of information, was easily determined when the S/N ratio was -25 db. Thus it appears that a communication system using this correlator would compare favorably with an ideal coding system.

[14] C. E. Shannon, "A Mathematical theory of communication," Bell Syst. Tech. Jour., vol. 27, pp. 379, 623; 1948.

VII. ACKNOWLEDGEMENTS

The assistance of many of the staff and graduate students of the electrical Engineering Department at the University of Wisconsin in the course of this work is gratefully acknowledged. In particular the assistance of Mr. T. M. Burford in the design and construction of the delay line was most valuable.

A WIDE-BAND SQUARE-LAW COMPUTING AMPLIFIER*

A. S. Soltes
Air Force Cambridge Research Center
Cambridge, Massachusetts

SUMMARY – Wide-band computing amplifiers capable of accurately and continuously yielding output signals with amplitudes proportional to the mathematical square of their input amplitudes have been developed. These computers are designed to be inserted in signal channels in much the same manner as conventional amplifiers, but impart thereto a square-law transfer characteristic. Details are given concerning a model that provides an accuracy of the order of 1 per cent of full scale over an output dynamic range of 40 db when operating on fractional microsecond, pulsed carrier signals.

I. INTRODUCTION

The square-law amplifier was developed to satisfy the need for a wide-band computing device which could accept a continuous stream of short, pulsed signals—such as might appear, for example, at the output of a radar receiver—and deliver them at its output with amplitudes proportional to the mathematical square of their input amplitudes, that is $E_{out} = KE_{in}^2$. An accuracy of 1 per cent of full scale was required over an output range of 40 db. The input signals were available in the form of a 40-mcps pulsed carrier with pulses as short as ½ microsecond.

A review of the existing methods for generating prescribed nonlinearities—such as the static characteristics of conventional vacuum tubes and crystals (alone or in networks) biased diodes, conventional square-law detectors and feedback function generators—revealed that all were inadequate for use in this application from the standpoint of either accuracy or bandwidth. A new approach was necessary to develop square-law devices with sufficiently accurate, dependably reproducible, wide-band characteristics.

2. DESCRIPTION OF THE SQUARE-LAW CIRCUIT ELEMENT[1]

The basic method employed is that of deflecting an electron sheet across suitably shaped target electrodes in a beam deflection tube (Fig. 1). This method is essentially inertialess and hence usable over a wide band of frequencies with an upper limit set only by transit time. The target used is parabolic in shape, this shape being extremely tolerant with regard to the geometry of the electron beam with which it must be combined to produce a square-law characteristic. Assuming that we have a beam that is uniform with height, and whose shape remains constant while it is linearly deflected across a parabolic target electrode, we will obtain a target current, i_{out} vs input voltage e_{in}, characteristic of the form

$$i_{out} = i_0 + k(e_0 + e_{in})^2 \text{ amp} \qquad (1)$$

(where i_0, e_0, and k are constant) regardless of the shape of the beam. Equation (1) represents a parabola with vertex at point e_0, i_0 with respect to the origin. The magnitude of the constants will vary from one beam shape to another, but the parabolic shape of the characteristic will remain invariant.

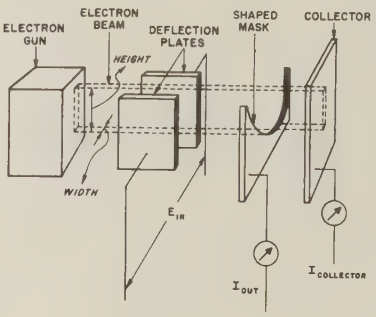


Fig. 1 – Functional schematic of beam deflection tube.

Tubes embodying this principle were built by the Raytheon Manufacturing Company on contract with this laboratory.[2] Several models with a common basic design were constructed representing varying degrees of electrical sensitivity and structural refinement. In general, tubes with higher sensitivities required more complicated electrode configurations and finer construction.

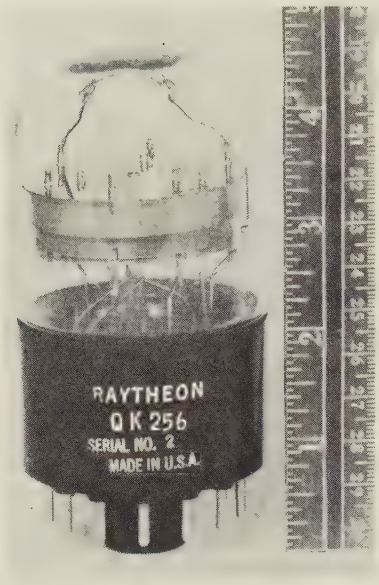
A successful amplifier was designed around a simple, though relatively insensitive, tube model designated QK-256 serial No. 2 and shown in Fig. 2.[3] As can be seen from the cut-away view (Fig. 2 (b), the tube is designed with cylindrical symmetry about an axially located cathode, the electrons traveling radially out from it in the form of a disk beam. The washer-like deflection plates located above and below the beam raise or lower its outer edge in accordance with the input voltage. The parabolic target is lying on its side in the form of a cylinder con-

* Originally published by the Electronics Research Division of the Air Force Cambridge Research Center, November, 1952.

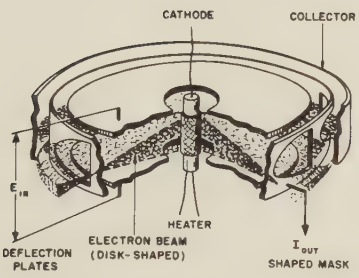
[1] This development is described in A. S. Soltes "A wide-band square-law circuit element," Air Force Cambridge Research Tech. Rept. S4-2, E5110, January, 1954.

[2] "Study of a beam deflection tube approach toward obtaining non-linear characteristics," Raytheon Mfg. Co. Quart. Prog. Repts., AF Contract No. AF 19(122) 17; December 28, 1948.

[3] Copies of this prototype have been designated QK-329.



(a)



(b)

Fig. 2 – (a) QK-256 beam deflection square-law tube; (b) QK-256 cut-away view.

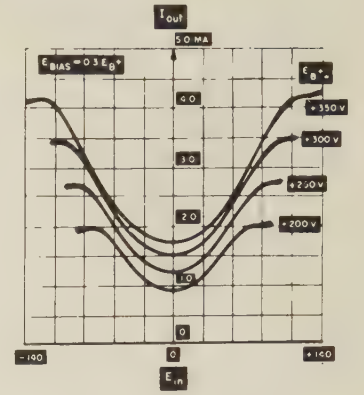
centric with the cathode. A collector ring surrounds the shaped mask to pick up those electrons not previously intercepted.

For mechanical convenience, the shaped target is divided into a number of identical small parabolas of convenient dimensions which are the equivalent of one long parabola. This method of construction also results in a reduction of distortion from certain types of imperfections that could occur in the fabrication of the tubes.

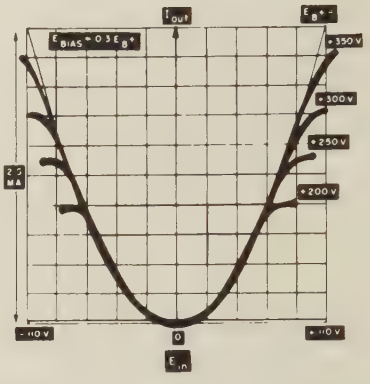
Static characteristics of the QK-256 serial No. 2, which are typical for this model, are shown in Fig. 3. The operating conditions are with the deflection plates biased at a constant fraction of $B+$. With this arrangement, wide variations in $B+$ produce but little change in curvature, except for the location of the overload point, and Eq. (1) is thus well approximated. The magnitude of k in Eq. (1) for these tubes average around 0.2 micromhos/volt.

3. GENERAL CONSIDERATIONS

The square-law amplifier is essentially a square-law circuit element surrounded by linear auxiliaries



(a)



(b)

Fig. 3 – (a) Static characteristics of QK-256; (b) comparison of curved portions of (a) with parabola.

which feed it and provide signals at the frequencies, impedances and levels required by the particular application. As such, its action amplitudewise maybe represented by the same expression as that for a square-law element alone, with constants incorporated to take account of the gains of the linear auxiliaries.

There are a number of ways of obtaining square-law operation from a square-law circuit element. Among them are the direct use of the static characteristic in a low-pass or video type of circuit; as a square-law detector with bandpass input and video output, or as a square-law amplifier with bandpass input and output.

For the particular application at hand, the input signal was available in the form of a pulsed carrier. Operation of the square-law tube as a video amplifier would offer no advantage over its use as a square-law detector, but would require that it be preceded by a linear detector. The discussion of the relative merits of different modes of operation will therefore be confined to amplifier vs detector.

If we assume a carrier input signal $e_{in} = E \cos \theta$ to a square-law tube with a static characteristic of the form of Eq. (1), the complete output current will be

$$i_{out} = \left(i_0 + ke_0^2 + \frac{kE^2}{2} \right) + 2ke_0 E \cos \theta + \frac{kE^2}{2} \cos 2\theta. \quad (2)$$

This output contains average, fundamental and second-harmonic components. Square-law detector operation would be obtained by selecting the average component of the output

$$(i_{out})_{avg} = i_0 + ke_0^2 + \frac{kE^2}{2} = \text{constant} + \frac{kE^2}{2}, \qquad (3)$$

and bandpass amplifier action would be achieved by selecting the second-harmonic component as the desired output

$$(i_{out})_{2\theta} = \frac{kE^2}{2} \cos 2\theta. \qquad (4)$$

The latter output (Eq. [4]) has the advantage of being independent of the location of the static characteristic with respect to the origin. It would therefore not require any centering voltage $-e_0$, and would be immune to possible error from shifts in i_0 or e_0 to which a direct-coupled detector output would be prone. As a result, the second-harmonic output could be expected to provide non-critical operation with a dynamic range extending down to the tube noise. On the other hand, stability of the "constants" might represent the lower limit of accuracy and, hence, dynamic range for the detector mode of operation. The inherent stability could probably be improved, if necessary, by means of circuitry, but this would increase the complexity of the device.

The second-harmonic output also offers certain conveniences in applications where it is desirable to preserve the signal in carrier form. Scale factor variations which might occur could be corrected by manual or automatic adjustment of the gain of associated linear amplifiers. A disadvantage in some applications (such as the one at hand) where it is desired that the signal frequency at input and output be the same, is that a linear frequency conversion is also necessary to compensate for the frequency doubling consequence of the squaring action. Both the scale factor adjustment and linear frequency conversion provisions may be satisfied, however, by using conventional techniques and circuits.

The second-harmonic amplifier type of square-law device operation was selected for the square-law amplifier on the basis of its potentially simple and uncritical performance capabilities, despite the necessary consequence of incorporating a linear frequency converter within the unit. The appropriate expression for the amplitude characteristic of the square-law amplifier is, therefore, that of the second-harmonic amplifier

$$e_{out} = K_{overall}\, e_{in}^2, \qquad (5)$$

where $K_{overall}$ is now the scale factor of the complete amplifier. This scale factor may be determined from the specified maximum input and output levels, i.e.,

$$K_{overall} = (e_{out})\ \text{max}/(e_{in})^2\ \text{max}. \qquad (6)$$

The scale factor $K_{overall}$ may be broken down in terms of the functional components of the square-law amplifier which are listed here for the typical case shown in the block diagram of Fig. 4. They are:

1. Linear converter (supplied with LO power from external source)
2. Driver amplifier
3. Square-law stage
4. Output amplifier.

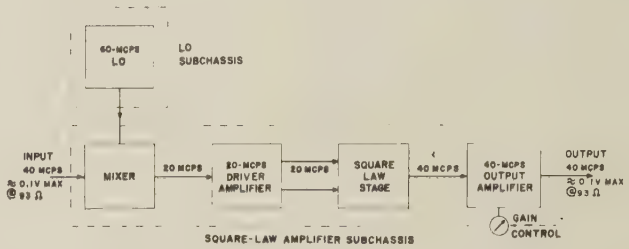


Fig. 4 — Functional block diagram of typical square-law amplifier.

Then

$$e_{out} = k_4\, k_3\, (k_2\, k_1)^2\, e_{in}^2, \qquad (7)$$

where the k's represent the gain constants of the functional components with subscripts numbered to correspond with the list above. By comparison of Eqs. (6) and (7),

$$K_{overall} = k_4\, k_3\, (k_2\, k_1)^2. \qquad (8)$$

It is apparent from the preceding equation that k_2k_1, the gain of the components preceding the square-law stage, is more important to the overall gain than k_4k_3, the gain of the square-law stage and output amplifier. In order to achieve a given value of $K_{overall}$, therefore, it would be desirable to make k_2k_1 as large as possible.

4. FACTORS IN THE DESIGN OF THE INDIVIDUAL FUNCTIONAL COMPONENTS

4.1 The Linear Converter

As was pointed out earlier, the linear converter is required only in those applications where the output center frequency must be the same as the input center frequency. In the case at hand, a 40-mcps input signal is provided, and a 40-mcps output is required. The linear downward frequency conversion necessary to compensate for the frequency doubling action of the square-law stage is accomplished at the very input, ahead of the squaring operation. The incoming 40-mcps signal is heterodyned down to 20 mcps with a 60-mcps LO signal in a linear mixer, in this case a 1N43 crystal. The mixer is operated at as high a level as is possible without damage to the crystal so that the ratio of signal to mixer noise will be kept at a maximum. The same overall result of 40 mcps at both input and output could, of course, be achieved by squaring the 40-mcps input signal and then heterodyning

the resulting 80 mcps down to 40 mcps with a 120-mcps LO and a mixer. The first method described is being used because it possesses certain advantages over the latter method including better S/N ratio and stability.

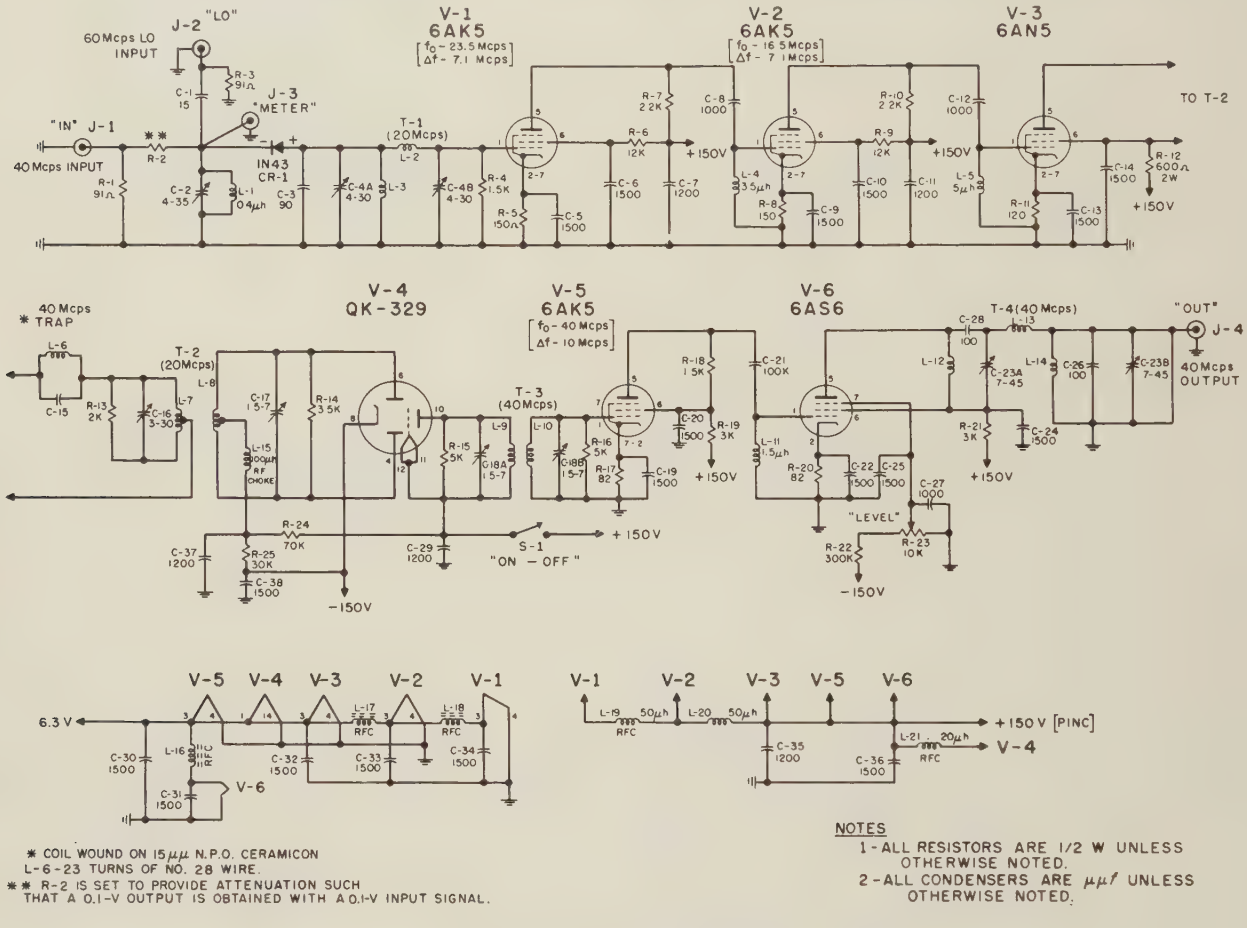
4.2 The Driver Amplifier

It is the purpose of the driver amplifier to step up the single-ended, 20-mcps signal from the mixer or input to the relatively high level capable of being handled by the square-law stage and to furnish this output in balanced form. (The type QK-329 as used in the present application is capable of accurately handling inputs up to about ±75 volts peak.) It is desirable to have as high a signal level as possible at the input to the square-law tube, because for the bandwidth required its output noise is significant. Furthermore, as was shown by Eq. (8), for a given $K_{overall}$ any loss in gain k_1k_2 would have to be compensated for by an increase in gain k_3k_4 equal to the square of the loss. The maximum gain k_1k_2, which can be used in the present design is limited by the signal level which can be handled linearly by the driver. Higher linear signal level capabilities could have been achieved in the driver at the expense of greater circuit complexity.

When a linear converter is used ahead of it, the driver amplifier also serves as a bandpass filter to reject all frequency components of the mixer output other than the desired stepped-down frequency of 20 mcps and its associated sidebands.

4.3 The Square-Law Stage

The Raytheon QK-329 beam deflection square-law tube is operated with a positive bias voltage which is a constant fraction of the plate supply voltage, and with the 40-mcps second harmonic of the input signal selected as the output signal. As mentioned earlier, this mode of operation has been found to yield a stable square-law response dependent only upon the curvature of the tube's static characteristics and essentially independent of power supply variations. No dc centering voltage to the deflection plates is employed. Magnetic as well as electrostatic shielding is provided for this stage to prevent its being influenced by stray magnetic fields.



* COIL WOUND ON 15 μμ N.P.O. CERAMICON L-6 – 23 TURNS OF NO. 28 WIRE.
** R-2 IS SET TO PROVIDE ATTENUATION SUCH THAT A 0.1-V OUTPUT IS OBTAINED WITH A 0.1-V INPUT SIGNAL.

NOTES
1 – ALL RESISTORS ARE 1/2 W UNLESS OTHERWISE NOTED.
2 – ALL CONDENSERS ARE μμf UNLESS OTHERWISE NOTED.

Fig. 5 – Circuit diagram of 40 mcps square-law computing amplifier.

4.4 The Output Amplifier

The output amplifier serves to select the 40-mcps second-harmonic component of the square-law stage output signal, filtering out other components such as fundamental and average. It amplifies this second-harmonic signal to the voltage and impedance levels specified for

e output signal. A gain control to adjust the proportion-ity constant of the overall square-law amplifier $K_{overall}$ located here so that it will not affect S/N or dynamic nge.

5. PERFORMANCE DATA

The complete circuit diagram of a model of a wide-and square-law computing amplifier which follows the nctional block diagram is given in Fig. 5. This ampli-er possesses the following measured overall character-tics:

Center frequency: 40 mcps at input and output.
Bandwidth: 5 mcps with respect to output.
Dynamic range: 40 db at output; 20 db at input.
Accuracy: within 1 percent of full scale.
Maximum levels: 0.1 volt at input and output.
Proportionality constant, $K_{overall}$: 10.
Input and output impedances: 93 ohms.

It can be seen from the bandwidth that signals as hort as a fraction of a microsecond can be handled by his model. Fig. 6 is a photograph of this wide-band quare-law computing amplifier.

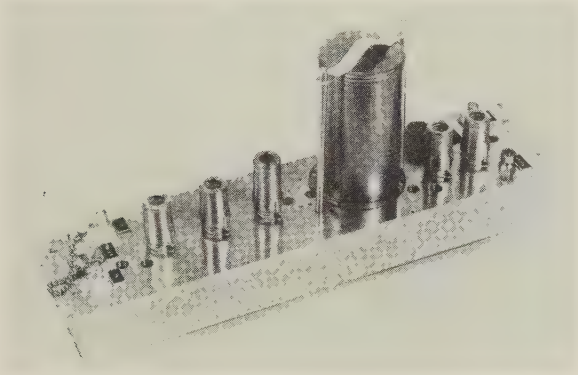


Fig. 6 — Photo of square-law computing amplifier.

6. LIMITATIONS IN PRESENT DESIGN

The dynamic range achieved in the above design, ver which the rated accuracy may be obtained, is bounded by noise originating in the square-law stage at the low end and by overload of the driver amplifier at the high end. Full use is not made, therefore, of the dynamic range available from the square-law stage. Although adequate for its intended application, the dynamic range could be made greater, if desired, within the capabilities of the existing square-law stage by the use of an improved driver amplifier. The one shown in the circuit of Fig. 5 is sufficiently linear up to an output level of less than 10 volts, whereas the square-law stage, as is, is able to handle inputs up to approximately 75 volts without over-loading. The use of degeneration, larger tubes and push-pull operation are several of the methods that might be employed to extend the linear range of operation of the driver amplifier. The dynamic range of the square-law stage itself could also be extended further by increasing the value of $B+$ applied to it.

7. CONCLUSIONS

Wide-band computing amplifiers have been developed which are capable of accurately and continuously yielding output signals with amplitudes proportional to the mathematical square of their input amplitudes. The beam deflection square-law tubes which were devised to meet the needs of such amplifiers, possess static characteristics which have unusually stable square-law curvature. Advantage has been taken of this feature by operating the tube as a second-harmonic amplifier, whose output is dependent only upon the stable curvature of the static characteristic.

8. ACKNOWLEDGMENTS

It is desired to give credit to B. C. Gardner[4] and R. M. Unger of the Raytheon Manufacturing Company who designed and constructed the beam deflection square-law tubes used in the equipment, and to acknowledge the helpful assistance of A. Moccia, G. Fine and T. Willson of this laboratory during the course of this development.

[4] Now with Varian Associates.

AN ANALOG MULTIPLIER USING THYRITE[1]

L. D. Kovach and W. Comley
Douglas Aircraft Company, Inc.
El Segundo, California

SUMMARY – An investigation into the use of thyrite as an inexpensive nonlinear element in the electronic analog computer indicates that this material has great value as a device capable of delivering an output voltage proportional to the square of the input voltage. The factors discussed are the characteristics of the material and the means by which these may be modified to produce a device capable of squaring with an accuracy of 1.25% from dc to frequencies in excess of 1000 cps. This ability to square makes possible the more important operation of the multiplication of two variable voltages.

INTRODUCTION

Many analog computers, particularly those of the differential analyzer type, which have been produced in recent years have been designed primarily for the solution of linear equations. To be sure, much has been done on many of these machines to include means for handling nonlinearities but not without considerable cost to the user. Considerations of economy often preclude the purchase of such items as servo and electronic multipliers, arbitrary function generators, etc. It is certain therefore, that any good economical means to extend the versatility of an otherwise linear machine is much to be desired.

This has been a particularly difficult problem on computers which operate to an upper frequency limit of 1000 cps since the cost of nonlinear devices rises rapidly as the pass-band is increased. The excellent characteristics of thyrite with regard to phase shift and frequency response make possible many nonlinear operations as the material imposes no restrictions on even the fastest type computers.

CHARACTERISTICS OF THYRITE

Thyrite is made by pressing silicon carbide with a suitable ceramic binder at high pressure followed by a firing operation at a high temperature. Electrical contact is made by means of a metal coating sprayed on the surfaces. Thyrite is a nonlinear resistor in which the current varies as a power of the applied voltage. An explanation of this behavior has been given by Schwertz and Mazenko.[2]

The expression which approximates the volt-ampere characteristic of a thyrite resistor is

$$I = K E^n, \tag{1}$$

where K is a constant which represents the amperes at one volt and n is the exponent. A preliminary investigation indicated that the most satisfactory results could be expected from the catalog no. 8396839G1 thyrite miniature resistors, particularly from those having an exponent between 2.49 and 2.86.[3] Thirty samples ordered on this basis were used in the present investigation.

In order to change the exponent to 2.0 it is necessary to place a linear resistance in series with the thyrite. This changes the volt-ampere characteristic[4] from that shown in equation (1) to

$$E = IR_0 + CI^{\frac{1}{n}} \tag{2}$$

where $C = 1/K^{\frac{1}{n}}$. Before the correct value of R_0 can be calculated the volt-ampere characteristic of the thyrite alone must be determined by means of the circuit shown in Fig. 1.

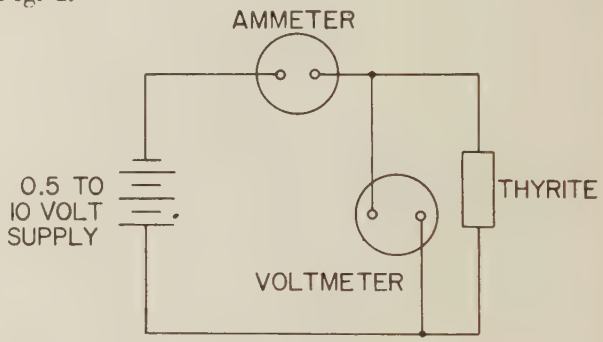


Fig. 1 – Circuit for determining E - I curve.

A calibrated vacuum tube voltmeter having an input impedance of 10 megohms and an 0.25% ammeter were used. The voltage was varied from 0.5 to 10 volts in increments of 0.5 volt and the current was recorded at each point. Results for a typical sample are shown in Fig. 2, together with a 10,000 ohm linear resistance for comparison.

ADJUSTMENT OF THE EXPONENT

Adjustment of the exponent to exactly two requires the calculation of the thyrite resistance at each point of the E-I curve. A plot of this resistance against current is shown as curve (A) in Fig. 3.

The general equation of the straight line (B) which is the best approximation to (A) is

$$-m \log I + \log R = a \tag{3}$$

[1] The name *Thyrite* is a registered Trademark of the General Electric Company.

[2] F. A. Schwertz, and J. J. Mazenko, "Nonlinear semiconductor resistors," *Jour. Appl. Phys.*, vol. 24, no. 8, pp. 1015-1024; August 1953.

[3] The commercial range of this exponent is from 2.49 to 3.36.

[4] Theodore Brownlee, "The calculation of circuits containing thyrite," Part I, *G. E. Rev.*, pp. 175-179; April, 1934.

ere m is the slope of the line and a is a constant. If a ear resistance R_0 is placed in series with the thyrite desired equation becomes

$$\log I + 2 \log (R + R_0) = b \qquad (4)$$

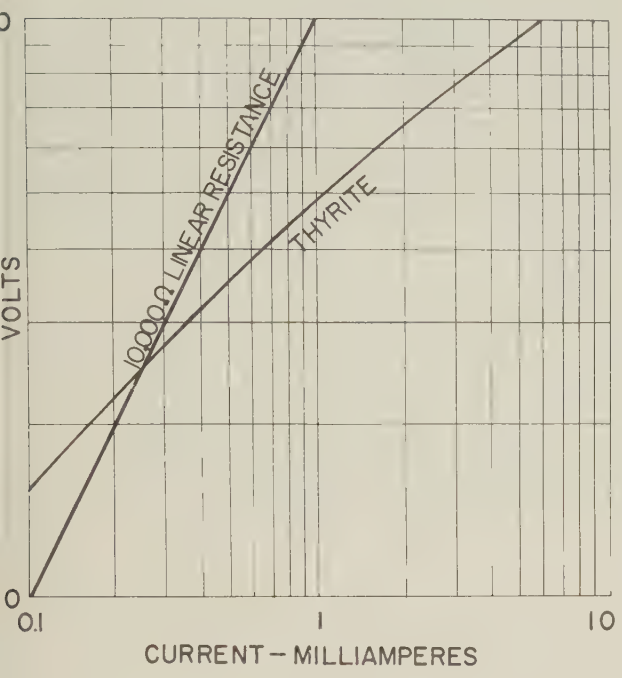


Fig. 2 – Typical E - I curve.

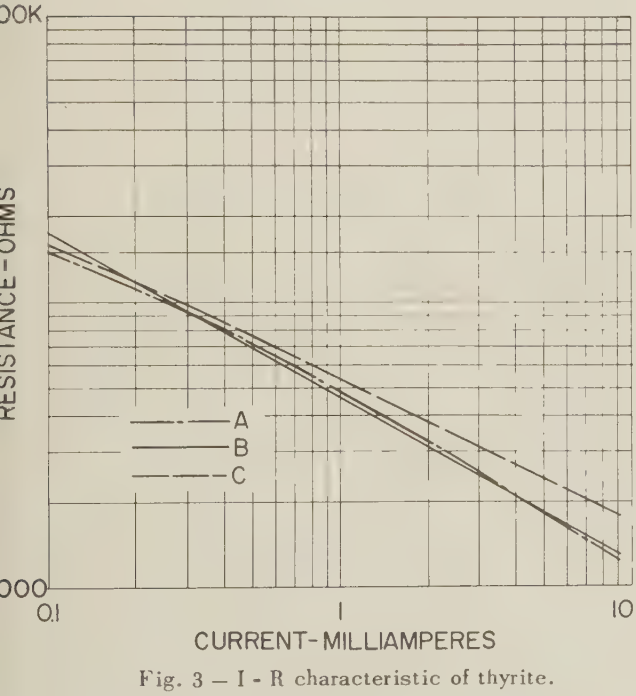


Fig. 3 – I - R characteristic of thyrite.

here b is a constant. By choosing two values of I and R om curve (B) of Fig. 3 it is possible to calculate R_0 om (4). Values of R_0 ranged from 253 to 862 ohms in the) samples tested. The addition of R_0 to curve (A) roduces curve (C) in Fig. 3. It should be noted that the ight downward concavity of curve (A) is a definite advantage since the addition of R_0 to the points of curve (A) produces a more nearly linear curve (C). A parabolic transfer function results if this curve is linear and has a slope of -0.5.

Comparison of input and output voltages by means of the circuit shown in Fig. 4 determines the degree of departure from this ideal transfer characteristic.

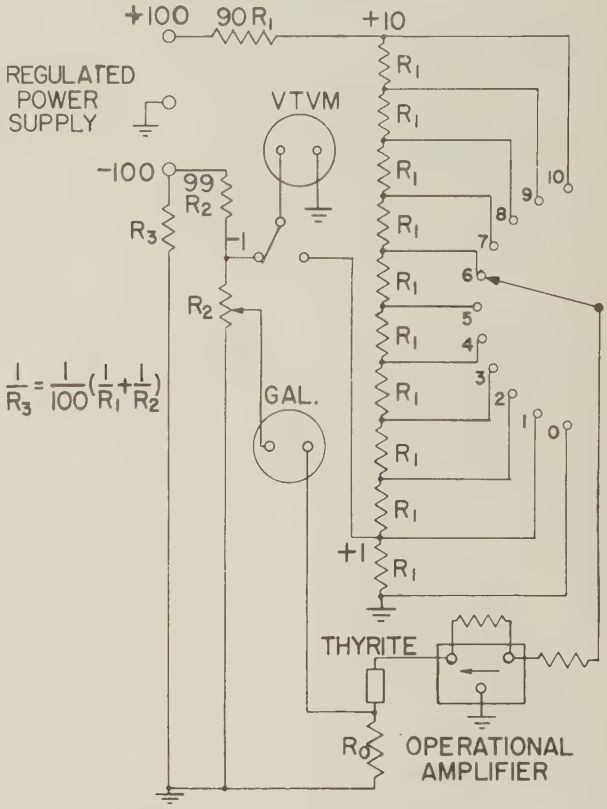


Fig. 4 – Circuit for accuracy check.

A null type of indication in a circuit containing precision resistors provides a greater accuracy than may be attained with a direct reading meter system. A ten-turn potentiometer makes it possible to read the output voltage to three significant figures. For convenience the insertion loss is increased to exactly 100 in the typical set of values of the following table:

INPUT VOLTS	OUTPUT VOLTS	ERROR
1	0.0112	+ 12.00%
2	0.0392	- 2.00%
3	0.0871	- 3.22%
4	0.1560	- 2.50%
5	0.2470	- 1.20%
6	0.3597	- 0.08%
7	0.4910	+ 0.20%
8	0.6400	0
9	0.8060	- 0.49%
10	0.9875	- 1.25%

Thyrite is sufficiently stable and unaffected by ordinary changes of temperature and humidity so that the above readings could be consistently repeated.

On the basis of measurements such as those described above it was found that the squaring device is capable of producing the square of a voltage with an accuracy of 1.25% of full scale. Of the 30 samples checked there were 11 whose accuracy was at least 1½% full scale and 2 whose accuracy was at least 1¼% full scale.

APPLICATION

As may be seen in Fig. 5, the anti-symmetrical nature of the thyrite results in an output whose sign is dependent upon the sign of the input voltage. Before progressing further to correct this condition, however, it should be noted that this characteristic is often encountered in the simulation of bidirectional physical systems such as the square law damper.

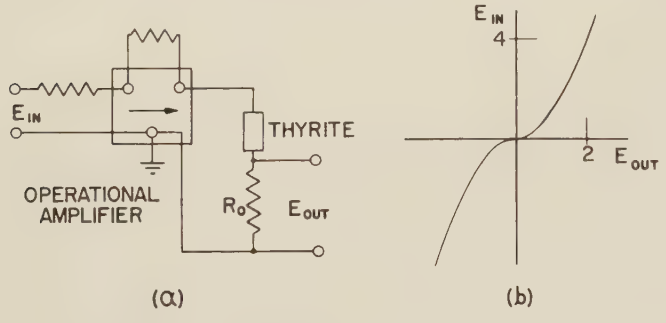


Fig. 5 – Anti-symmetrical squaring circuit.

In order to provide a symmetrical square the sign of the output must always be positive regardless of the sign of the input voltage. This is accomplished as indicated in Fig. 6. Germanium diodes are used to provide an absolute value input to the thyrite. Should a square of the opposite sign be required, it may be obtained by the

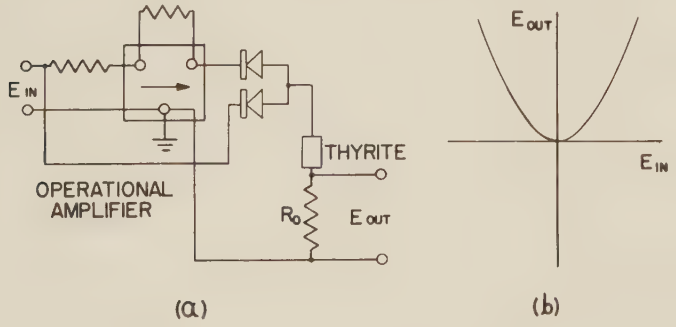


Fig. 6 – Symmetrical squaring circuit.

simple expedient of reversing the diodes. For convenience the diodes, the correct linear resistance and the thyrite are mounted on a small insulating panel provided with banana plugs. One of these panels is shown in Fig. 7. The panel is plugged into the output terminals of an operational dc amplifier.

In order to operate the thyrite over its most accurate range the full scale voltage input should provide a current of 10 ma through R_0. The insertion loss of the squaring operation is approximately 20. Depending upon the requirements of the problem, a subsequent amplification by the same factor may be required.

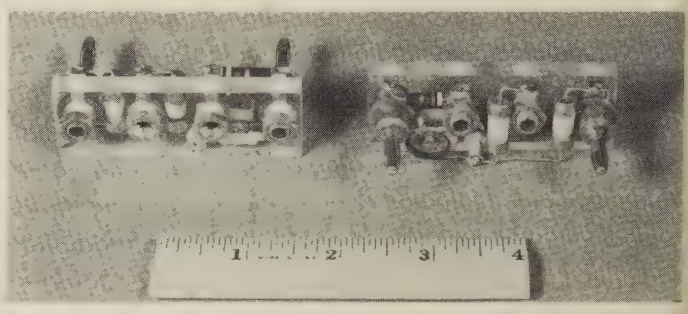


Fig. 7 – Thyrite squaring device.

It may be found that in obtaining the maximum accuracy from a given sample better results are achieved if the gain is adjusted to provide a match at some point other than full scale. Assuming a full scale input of 10 volts, setting the gain to provide the correct output for, say, 7 v. often results in reducing the error over the entire range. Under these conditions it may be necessary to consider 8 or 9 v as the maximum allowable input in order to realize the 1.25% full scale accuracy.

Possibly the most useful application is the use of the squaring circuit to perform the multiplication of two variables. Given the identity

$$(x + y)^2 - (x - y)^2 = 4xy \qquad (5)$$

it may be seen that a combination of squaring and addition operations will provide the product. These are shown schematically in Fig. 8.

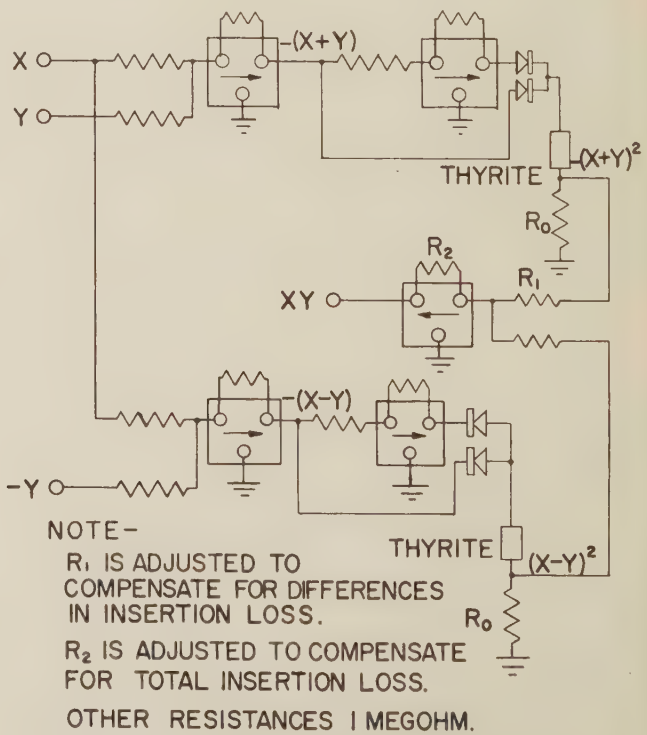


Fig. 8 – Circuit for multiplication.

It should be remembered that in multiplication the aximum input to either x or y is one-half the allowable put for a squaring operation alone since the identity

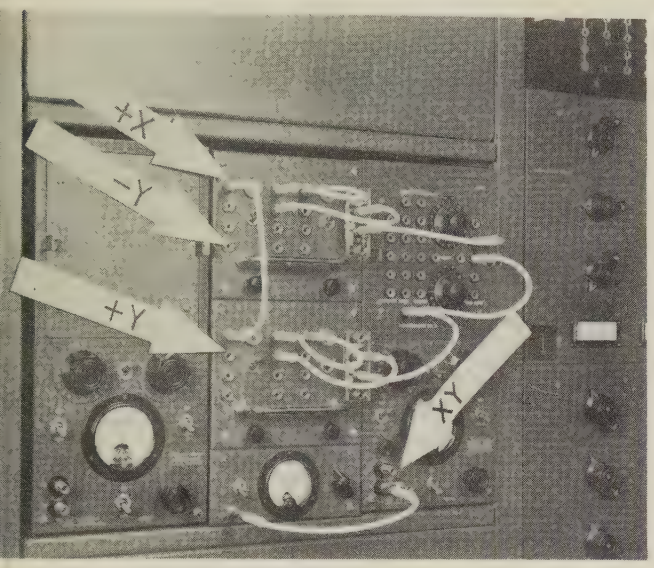


Fig. 9 – Thyrite multiplier

volves squaring the *sum* of x and y. This value should ot exceed the full scale voltage of the squaring opera- on. The accuracy of this method of multiplying should limited only by the accuracy of the squaring operations d the linearity of the amplifiers. Fig. 9 depicts the tual equipment connected to perform a multiplication.

CONCLUSIONS

A device consisting of thyrite whose exponent has been properly adjusted is capable of squaring voltages with an accuracy of 1.25% of full scale. When operated within its proper range the device is not subject to drift and at 1000 cps it will introduce less than one degree of phase shift. It is estimated that a multiplier using this squaring principle can be built for less than one-half the cost of present electronic multipliers having the same frequency response. Finally, ease of adjustment and simplicity of the circuit results in a multiplier lacking in many of the complications found in similar devices.[5,6,7,8,9]

[5] E. J. Angelo, Jr., "An electron-beam tube for analog multiplication. *MIT Tech. Rept. no. 249*, October 27, 1952.

[6] Samuel E. Dorsey, "An Electro-Mechanical Multiplier for Analog Computer Application," *Proc. Elec. Comp. Symp.* Los Angeles, California (sponsored by IRE); April 30, 1952.

[7] Edwin A. Goldberg, "Step multiplier in guided missile computer," *Electronics*, pp. 121-124; August, 1951.

[8] Edwin A. Goldberg, "A high-accuracy time-division multiplier," *RCA Rev.*, vol. XIII, no. 3, pp. 265-274; September, 1952.

[9] Byron O. Marshall, Jr., "An analogue multiplier," *Nature*, pp. 29-30; January 6, 1951.

A SUB-AUDIO TIME DELAY CIRCUIT

C. D. Morrill
Goodyear Aircraft Corporation
Akron, Ohio

SUMMARY – Through the use of an electronic differential nalyzer arranged to give a sixth-order approximation of the aplace shift operator, it is possible to reproduce an input ignal and delay it T seconds when the highest angular fre- uency present in the input signal does not exceed 12/T. accuracy due to nonlinear phase shift is less than 2% under ese conditions. This device requires only linear computing ements, and will permit delays of seconds within a finite equency spectrum. An application to a closed loop industrial ontrol problem is cited and other applications are suggested.

INTRODUCTION

This paper describes procedures used in the development of a time-delay device capable of delaying electrical signals for time intervals of the order of seconds. The initial requirement was for a uniform time delay of one second for signals from dc to about ½ cycle per second. The method used permitted an accurate delay of one second for signals out to about two cycles per

second. It was found possible to make the time delay continuously variable and to produce delays of 60 seconds or more provided that the time delay, T, is equal to or less than $12/\omega$, where ω is the highest frequency of importance. Longer delays are possible with correspondingly increased complexity. The delay circuit was formed on the GEDA,* an electronic differential analyzer, and was used in the study of a simple process control characterized by dead time.

METHODS

As shown by Gardner and Barnes,[1] if:

$$L\ [\,e\,(t)\,]\ = E\,(s), \tag{1}$$

then

$$L\ [\,e\,(t-T)\,]\ = E\,(s)\ \epsilon^{-sT} \tag{2}$$

ϵ^{-sT} is called the Laplace shift operator. If $i\omega$ is substituted for s and the resulting function plotted on the complex plane, a unit circle centered at the origin is described; this is the locus of a uniform, loss-free transmission line. Thus, three methods of delaying an electrical signal are suggested:

1. The construction or simulation of an artificial transmission line; a synthetic approach to the problem.
2. The use of mathematical approximations to the exponential; an analytic approach to the problem.
3. A combination of these two methods.

In using the first approach, we may consider two types of networks – the ladder and the lattice. Prototypes of these networks are shown in Figs. 1 and 2. In this

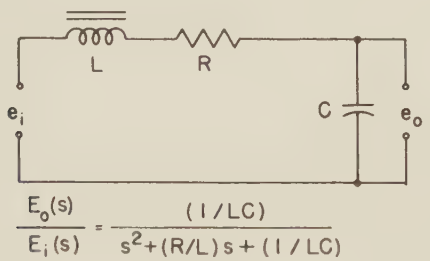


Fig. 1 – Prototype ladder network.

discussion, it is assumed that the networks are unterminated. As additional sections are required, they are coupled by means of impedance transformers having infinite input impedance, zero output impedance, and a unity transmission.

The transfer function of the network shown in Fig. 1 is:

$$\frac{E_0\,(s)}{Ei\,(s)} = \frac{(1/LC)}{s^2 + (R/L)\,s + (1/LC)} \tag{3}$$

The transfer function of the network shown in Fig. 2 is:

$$\frac{E_0(s)}{Ei(s)} = \frac{s^2 - (R/L)\,s + (1/LC)}{s^2 + (R/L)s + (1/LC)} \tag{4}$$

As the damping ratio $\frac{R}{2}\sqrt{\frac{C}{L}}$ of the characteristic equation of (3) and (4) is varied from a small to a large value, the shape of the initial portion of the phase shift characteristic changes from concave downward to concave upward; in the vicinity of 0.7, the phase shift characteristic is nearly linear. This characteristic is shown quite well by Brown and Campbell.[2]

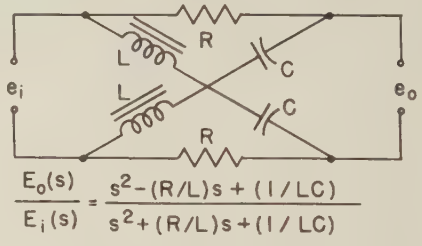


Fig. 2 – Prototype lattice network.

The phase shift of the low-pass network, Fig. 1, is given by:

$$\phi = \tan^{-1}\left(\frac{RC\omega}{1 - LC\omega^2}\right) \tag{5}$$

while that of the all-pass network, Fig. 2, is given by:

$$\phi = 2\tan^{-1}\left(\frac{RC\omega}{1 - LC\omega^2}\right) \tag{6}$$

Thus, since the arguments are identical, the all-pass network is twice as effective as the low-pass network for the intended purpose. As will be shown later, this 2:1 improvement requires only a small amount of additional analog computing equipment. An electrical network made up of L's, R's and C's becomes unreasonable for delays on the order of seconds.

The second or analytic approach involves the application of a series approximation of ϵ^x. Perhaps the best known such series is the Taylor series where:

$$\epsilon^x = 1 + X + \frac{X^2}{2!} + \frac{X^3}{3!} + \text{----} \tag{7}$$

*Goodyear Electronic Differential Analyzer (T.M. Goodyear Aircraft Corporation, Akron, Ohio).

[1] M. F. Gardner and J. L. Barnes, *"Transients in Linear Systems,"* John Wiley and Sons, Inc., New York, N.Y., p. 120, 1950.

[2] G. S. Brown and D. P. Campbell, "Principles of Servomechanisms", John Wiley and Sons, Inc., New York, N.Y., p. 100; 1948.

As will be shown, this series is inferior to a partial tion expansion given by Perron.[3] The Padé approximastates that:

$$\epsilon, \nu(X) = 1 + \frac{\nu X}{(\mu+\nu)!} + \frac{\nu(\nu-1)X^2}{(\mu+\nu)(\mu+\nu-1)2!} + \cdots + \frac{\nu(\nu-1)\cdots 2\cdot 1\cdot X^\nu}{(\mu+\nu)(\mu+\nu-1)\cdots(\nu+1)\mu!} \quad (8)$$

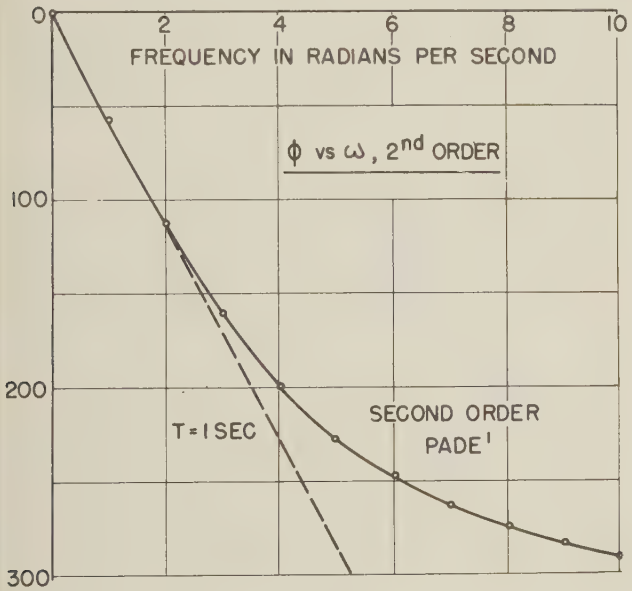
if:

$$\mu, \nu(X) = 1 - \frac{\mu X}{(\mu+\nu)!} + \frac{\mu(\mu-1)X^2}{(\mu+\nu)(\mu+\nu-1)2!} - \cdots + \frac{(-1)^\mu \mu(\mu-1)\cdots 2\cdot 1\cdot X^\mu}{(\mu+\nu)(\mu+\nu-1)\cdots(\nu+1)\mu!} \quad (9)$$

n:

$$\lim_{(\mu+\nu)\to\infty} \frac{F\mu,\nu(X)}{G\mu,\nu(X)} = \epsilon^x \quad (10)$$

$X = -ST$ and $\mu = \nu$, the resulting function of s repreits an all-pass network or group of networks in tandem ilar to the prototype shown in Fig. 2. When $\mu = \nu = 2$, becomes:



g. 3 – Phase characteristic of the second order Padé approximation to e^{-st}

. Perron, "Die Lehre Von Den Kettenbruchen," Chelsea ublishing Co. New York, N. Y., p. 459; 1950.

o the author's knowledge, the Padé approximation to ϵ^x was rst used to produce a time delay by Emory Lakotos, formerly ith the Bell Telephone Laboratories, Murray Hill, New rsey.

$$\epsilon^{-sT} = \frac{1 + \frac{2}{4}(-sT) + \frac{2}{4\cdot 3\cdot 2}(S^2T^2)}{1 - \frac{2}{4}(-sT) + \frac{2}{4\cdot 3\cdot 2}(S^2T^2)} \quad (11)$$

$$= \frac{S^2T^2 - 6sT + 12}{S^2T^2 + 6sT + 12} \quad (12)$$

$$= 1 - \frac{12sT}{s^2T^2 + 6sT + 12} \quad (13)$$

Fig. 3 shows phase shift as a function of frequency for (13). It is the equivalence of (12) and (13) which permits the use of the doubly effective all-pass network of Fig. 2 with only a slight increase in computing elements.

Fig. 4 is a schematic diagram of the computer circuit required to solve (13). It should be noted that variable delays may be produced by connecting the inputs of the

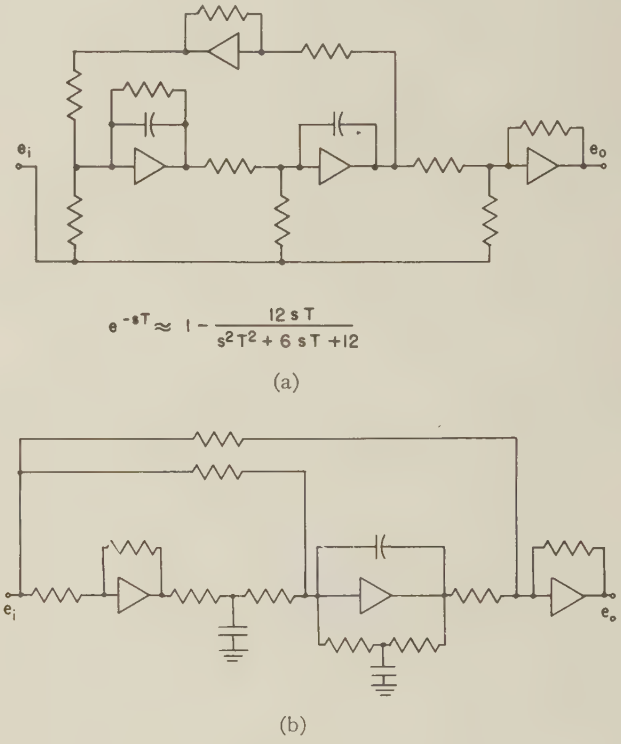


Fig. 4 – (a) Schematic diagram for the solution of (13). (b) Simplified circuit.

integrators to a set of ganged potentiometers, the position of which may be made proportional to the required time delay. When only a constant delay is required, the circuit of Fig. 4(a) (and Fig. 8 in the Appendix) may be simplified as shown in Fig. 4(b); usually amplifier 1 and/or 3 of Fig. 4(b) may be eliminated since they may be part of the principle computing circuit.

The effectiveness of a sixth order Taylor series may be compared to that of a sixth order partial fraction

expansion in Fig. 5. Increasing the number of terms in the Taylor series does not improve its quality due to lack of convergence. Furthermore, the Taylor series is not readily amenable to solution on a differential analyzer.

The fourth order Padé approximation was plotted in the same way; s was replaced by $i\omega$ and the phase shift was calculated. It was noticed that, although the fidelity of the second and fourth order approximations was quite good, the sixth order approximation deviated considerably from the desired linear characteristic. This result suggested the third approach – the synthesis of a "mop-up" delay equalizer which, when used with networks represented by the fourth order partial fraction expansion, would more nearly follow the desired straight line characteristic.

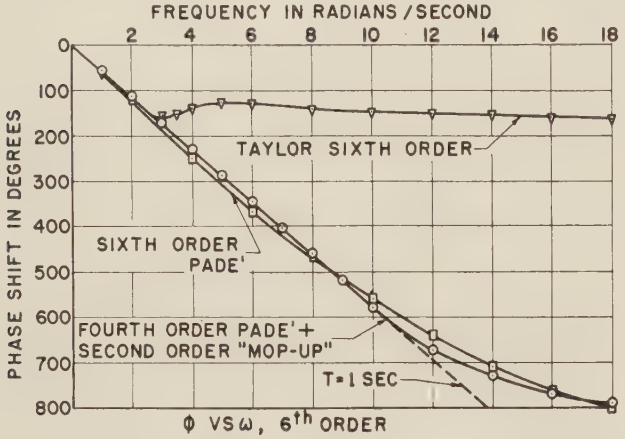


Fig. 5 – Effectiveness of various delay-time methods compared with ideal characteristic.

The "mop-up" equalizer was designed by:

1. Plotting the phase shift characteristic of fourth order Padé approximation to $\epsilon^{-i\omega T}$;
2. Plotting an error curve determined by the difference between the above curve and an arbitrary straight line of greater slope; and
3. Determining the network having a phase characteristic which most nearly followed the error curve.

Fig. 5 shows the improvement that may be obtained by employing a combination of synthetic and analytic methods. This combination is capable of an accuracy of 2% up to a frequency-delay product of 12 radians. As indicated in the block diagram, Fig. 6(a), three quadratic circuits are employed. Each ratio reduces to an expression of the form

$$1 - \frac{\gamma s T}{s^2 T^2 + \beta s T + \gamma}.$$

As noted before, this form of the expansion accounts for the economy of computing equipment using the all-pass version of the delay network; it is used in designing the equivalent circuit and in computing parameters. The computer circuit and a method of calculating constants are included in the appendix.

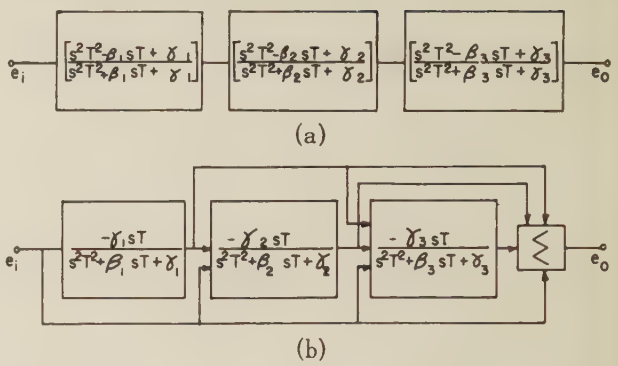


Fig. 6 – (a) Block diagram of time delay circuit. (b) Equivalent circuit.

EXAMPLE

In making tests of the circuit, a delay of four seconds was chosen. The action of the circuit is illustrated in Fig. 7, a simultaneous recording of a complex input waveform and the corresponding output. It will be noticed that there is a transient in the output prior to the start of the delayed signal. This transient rapidly decays, and does not subsequently affect the results.

It is interesting to note that this delay of four seconds corresponds to the delay (without loss) of a 19 ga. cable some 200,000 miles in length.

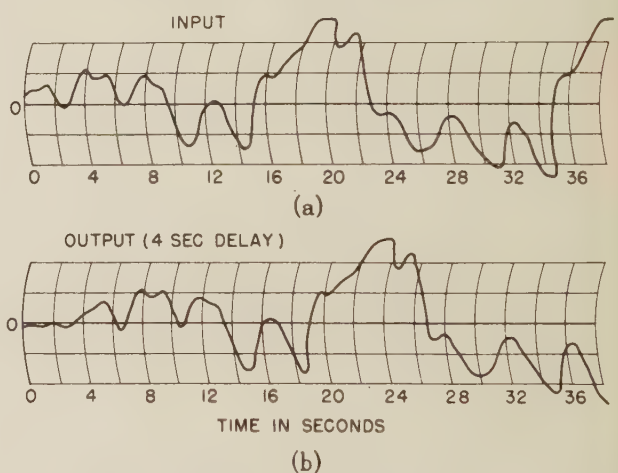


Fig. 7 – (a) Complex waveform input to delay network. (b) Delayed output.

CONCLUSIONS

It has been shown that relatively simple circuits may be used to delay electrical signals; these circuits may be used for delays ranging from less than 0.1 second to, perhaps, 100 seconds. It has also been shown that there is a sixth order partial fraction expansion, arrived at by semi-graphical means, which better approximates ϵ^{-sT} than the sixth order Padé approximation.

Although this method is capable of greater accuracy than conventional magnetic recording means of delaying signals, it is not capable of large values of ωT. Its

dvantage lies in its simplicity and in the fact that it ses equipment identical to that already being used in losed loop stability studies for which it usually provides, or a given delay, more than adequate bandwidth. It is robably a good deal simpler than digital storage would e; it has been estimated that, for similar results, a emory capable of storing about 12,000 binary digits ould be required.

The circuit described has been used in the analysis f an industrial process control[4] and in the study of the ynamics of the human operator.[5] It should be useful in ne study of certain diffusion problems and, perhaps, in ne study of the use of digital computers in closed loop ontrols.

APPENDIX

The parameters of the second and fourth order delay ircuits are readily obtained. If a sixth or higher order pproximation to ϵ^{-sT} is required, the circuit of Fig. 8 may be used. The circuit constants may be obtained from Table I. In using this table, let ωs be some average frequency to be passed through the delay network and let $\theta = \omega s T$. In general, the capacitors should be 0.1, 1.0 and 10.0 microfarads for delays of tenths, units, and tens of seconds. If every resistor does not lie between about 10K ohms and 10M ohms, the values of the capacitors should be adjusted.

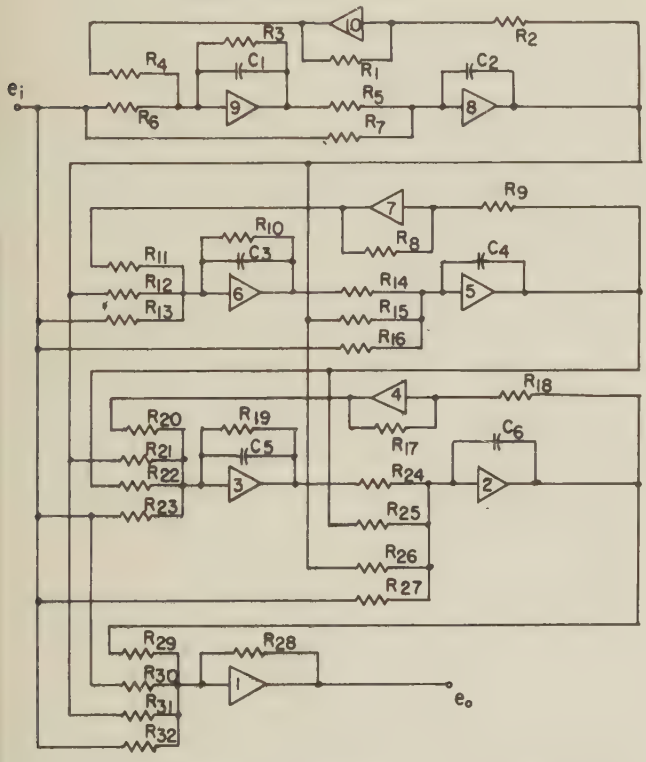


Fig. 8 — GEDA schematic for simulation of all-pass delay network represented by a sixth-order partial fraction expansion.

TABLE I

RESISTOR VALUES FOR SIXTH-ORDER TIME-DELAY NETWORK

$$R_1 = 3.590 R_2 \theta^{-1/3} (\theta^2 + 170)^{-1/6}$$

$$R_3 = 0.07673 \frac{I}{C_1}$$

$$R_4 = 0.2785 \frac{I}{C_2} \theta^{1/3} (\theta^2 + 170)^{-1/3}$$

$$R_5 = 0.2785 \frac{I}{C_2} \theta^{-2/3} (\theta^2 + 170)^{1/6}$$

$$R_6 = 0.01057 \frac{I}{C_1} \theta^{2/3} (\theta^2 + 170)^{-1/6}$$

$$R_7 = 0.03836 \frac{I}{C_1}$$

$$R_8 = 3.874 R_9 \theta^{-1/3} (\theta^2 + 89.6)^{-1/6}$$

$$R_{10} = 0.1056 \frac{I}{C_3}$$

$$R_{11} = 0.2581 \frac{I}{C_3} \theta^{1/3} (\theta^2 + 89.6)^{-1/3}$$

$$R_{12} = R_{13} = 0.02162 \frac{I}{C_3} \theta^{2/3} (\theta^2 + 89.6)^{-1/6}$$

$$R_{14} = 0.2581 \frac{I}{C_4} \theta^{-2/3} (\theta^2 + 89.6)^{1/6}$$

$$R_{15} = R_{16} = 0.05282 \frac{I}{C_4}$$

$$R_{17} = 5.119 R_{18} \theta^{-1/3} (\theta^2 + 41.4)^{-1/6}$$

$$R_{19} = 0.1554 \frac{I}{C_5}$$

$$R_{20} = 0.1953 \frac{I}{C_5} \theta^{1/3} (\theta^2 + 41.4)^{-1/3}$$

$$R_{21} = R_{22} = R_{23} = 0.06185 \frac{I}{C_5} \theta^{2/3} (\theta^2 + 41.4)^{-1/6}$$

$$R_{24} = 0.1953 \frac{I}{C_6} \theta^{-2/3} (\theta^2 + 41.4)^{1/6}$$

$$R_{25} = R_{26} = R_{27} = 0.0770 \frac{I}{C_6}$$

GER-4689, "Simulation of a Process Controller," Goodyear Aircraft Corporation, Akron, Ohio; March 11, 1952.

R. Mayne, "Some engineering aspects of the mechanism of body control," *Elec. Eng.*, March, 1951.

CONTRIBUTORS

HAROLD BELL, Jr. (S '52 – A '54) was born in in Milwaukee, Wis., on March 30, 1927. After serving two years in the U.S. Navy, he attended the University of Wisconsin and received the B.S., M.S., and Ph.D. degrees in electrical engineering in 1950, 1951, and 1953 respectively.

At present Dr. Bell is employed in the Aerophysics Laboratory of North American Aviation.

Dr. Bell is a member of sigma Xi, Tau Beta Pi, and Eta Kappa Nu.

WILLIAM COMLEY was born in Indianapolis, Indiana, on September 14, 1921. After completing the electronics curriculum at Los Angeles City College, he served as a communications officer with the U.S. Air Force for three years and was for the greater part of this time in charge of a number of high-powered ground transmitting stations.

In 1947 Mr. Comley, joined the engineering staff of Lear, Inc. of California as a design engineer in aircraft radio. He later served as project engineer for that company's line of phonograph pickups and wire recorder heads. For the past four years he has been employed by the Douglas Aircraft Company where he is currently engaged in the design of analog computing devices and auxiliary equipment for existing facilities.

LADIS D. KOVACH (A '53) was born on November 21, 1914 in Budapest, Hungary. He received a B.S. degree in Physics in 1936 and an M.S. degree in Mathematics in 1948, from Case Institute of Technology, an M.A. degree Education from Western Reserve University in 1940 and a Ph.D. in Mathematics from Purdue University in 1951.

From 1936 to 1948 he was associated with Picker X-Ray Corporation, American Shipbuilding Company and Ohio Crankshaft Company as electrical designer. He was instructor in Mathematics from 1946 to 1951, first at Case, then at Purdue. Since 1951 he has been with the Douglas Aircraft Company where he is at present a computing specialist.

Dr. Kovach is a member of Sigma Xi, the American Mathematical Society and an associate of the AIEE.

For a biography of ALAN L. LEINER see p. 21 of EC-3, No. 1, March, 1954.

MELVIN E. MARON was born in Bloomfield, New Jersey, on January 23, 1924. He attended the University of Nebraska where he received a B.S. in Mechanical Engineering and a B.A. in Physics. From 1947 to 1951 he attended UCLA and graduated from UCLA in June 1951 with a Ph.D. in Philosophy.

Dr. Maron joined the UCLA faculty and taught Logic and Philosophy of Science during the academic year 1951-52. In July of 1952, Dr. Maron became a member of the staff of the IBM Research and Development Laboratory in San Jose, California, and since then he has been engaged in a study of the mechanization of logic and the automatic handling of non-numerical information.

For a biography of MAURICE A. MEYER see p. 21 of EC-3, No. 1, March, 1954.

CHARLES D. MORRILL (A '48 – M '52) was born in St. Louis, Missouri, on October 31, 1919. He graduated from the University of Illinois with a B.S. degree in Electrical Engineering in 1941. From 1941 through 1945, he served in the Army Air Forces as a communications and radar officer in the United States and the Central Pacific Area. Between 1946 and 1948, Mr. Morrill was with the Long Lines Division of the American Telephone and Telegraph Company, Chicago, Illinois.

In 1948, Mr. Morrill joined the Aerophysics Department of the Goodyear Aircraft Corporation, Akron, Ohio, where he has since been primarily engaged in the development, design and application of electronic differential analyzers.

Mr. Morrill is a member of the Institute of Radio Engineers and Eta Kappa Nu.

WILLIAM A. NOTZ (A '53) was born in Washington, D.C. on January 15, 1922. He received the B.S. degree in Electronic Physics from Harvard University in 1943.

From 1943 to 1946 Mr. Notz served in the Army Signal Corps as a Radar Officer.

In 1948 he joined the NBS Electronic Computers Laboratory where he participated in the SEAC and DYSEAC development programs. In connection with the SEAC he conducted component research on magnetic materials. On the DYSEAC, he participated in developing the detailed logical design for several major units of the system, and particularly was responsible for maintaining over-all design compatibility between the various units of the system.

Mr. Notz is a member of the Association for Computing Machinery.

VINCENT C. RIDEOUT (M '44 – SM '53), was born Alberta, Canada, on May 22, 1914. He received the S. degree in engineering physics from the University of berta in 1938, and the M.S. degree in electrical engiering from the California Institute of Technology in 40.

From 1939 to 1946 Mr. Rideout was a member of the chnical Staff of the Bell Telephone Laboratories, ere he was engaged in microwave radio and radar search. In 1946 he joined the staff of the University of sconsin as an assistant professor of electrical engiering, and was appointed as an associate professor in 48. At present his work at the University of Wisconsin als principally with communication networks and alog computers.

Professor Rideout is a member of Sigma Xi.

J. LYNN SMITH was born in Floyds Knobs, Indiana, October 2, 1921. He received the A.B. degree in stronomy from Indiana University in 1942, and the M.A. gree in Astronomy from Indiana University in 1948.

Mr. Smith was a member of the U.S. Naval Observary from 1942 through 1944. He served in the U.S. Armed orces during World War II. Between 1948 and 1951 Mr. ith worked in photoelectric photometry at the Lick bservatory of the University of California. In 1951, he ined the NBS Electronic Computers Laboratory, where worked on the logical design of several classified mputers. On the DYSEAC, he participated in developing e detailed plans for several of the major units in the stem, particularly those concerned with input-output d manual control.

Mr. Smith is a member of the Association for Computing Machinery and the American Astronomical Society.

AARON S. SOLTES (S '45 – A '47), was born August 17, 1918 in New York, N.Y. He received the B.S. degree in physics from the City College of New York in 1939, and the M.S. degree in communication engineering from the Harvard Graduate School of Engineering in 1940.

He was a radio engineer at the Emerson Radio and Phonograph Corporation and the U.S. Army Signal Corps until 1944 when he went on active duty in the U.S. Navy as an electronics officer. After his release from the Navy in 1946, he joined the Air Force Cambridge Research Center and is presently assigned to the Computer Laboratory of its Electronics Research Directorate. He has been engaged in work on radar receiver-computer-system problems, and is in charge of a research project concerned with applications of nonlinear circuit elements and their development.

ARNOLD WEINBERGER (S '49 – A '50), was born on October 23, 1924. He received the B.E.E. degree from City College of New York in 1950, after serving in the Armed Forces.

In 1950 he joined the NBS Electronic Computers Laboratory, where he participated in the SEAC and DYSEAC programs. On the SEAC, he carried out engineering work concerned with the expansion and operation of the machine. On the DYSEAC, he participated in the development of the logical design of the system, particularly the arithmetic circuitry and the detailed wiring plans from which the machine was constructed.

Mr. Weinberger is a member of the Association for Computing Machinery.

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

REVIEW SECTION

It is the intention of this section to review articles that have been published since January 1, 1954, and to publish eventually reviews of all books pertaining to the computer field. All articles and books reviewed are numbered sequentially for each year; where known, the Universal Decimal Classification number is also given. The editors wish to express their gratitude to the reviewers who, through their efforts, make this section possible.

H. D. Huskey, Editor

GENERAL

54-57

Electronic Digital Computers — Conence in the United States — *(Nature,* . 171, p. 599-600; April 4, 1953.) This cle outlines the organization of the t AIEE-IRE computer conference comtee and gives the subjects to be ered at future conferences. The first these conferences, Philadelphia, cember 10-12, 1951, is described efly. At this conference a number of ge scale digital computers were deibed. The papers and discussion sented at this conference have been blished as *A Review of Electronic ital Computers: Joint AIEE - IRE mputer Conference,* and is available n AIEE.

D. E. Hart

1.375.2 54-58

Influence of Programming Techniques the Design of Computers — Grace M. pper and John W. Mauchly. (*Proc. .E.*, vol. 41, pp. 1250-1254; October, 53.) The authors begin by pointing out t a programmer is a kind of engineer o uses mathematical equations, flow arts, instruction codes, etc. as the ls of his trade. It is the job of the mputer designer to facilitate the task the programmer. The developments of w techniques in programming may have profound an influence on computer sign as would be produced by an enely new type of memory or switching ement. The authors go on to give a ncise account of the historical develment of programming techniques and the developments now taking place in ir organization. They give their views the influence which these techniques likely to have on the design of both ge and small computers.

M. V. Wilkes

54-59

Electron Tube Performance in Some pical Military Environments — D. W. arp. (*Elec. Eng.*, vol. 73, pp. 233-238; rch 1954.) The data presented here s been chosen from the mass of inmation assembled by ARINC through ld surveillance of equipments in tual military use. The methods of llecting data are described. Tube performance in land-based fixed communications and ship-borne equipments is described (airborne equipment has not accumulated enough hours of operation for the data to be useful). Graphs giving distributions of defects, rates of failure, g_m decay, cathode interface resistance, and so forth, are presented and explained. It is pointed out that the regularity of some of these plots should make it possible to predict failure rates after a relatively short testing period on a given tube in a given environment.

H. T. Larson

54-60

An Automatic Data Recording Camera Richard S. Hill. (*Rev. Sci. Instr.*, vol. 24, pp. 1001-1002; Oct., 1953.) This article reports the development of a 19-lb shutterless 35-mm camera with a film-advancing sprocket and a timer switch which fires a flash unit. The film advances at one frame per second and the flash unit fires once per second. The camera must operate within a darkened enclosure but this disadvantage is outweighed by low power consumption and versatility. The readings of any variety of meters, indicators, etc. may be recorded with this mechanism.

Arthur Dowling

621.375.2 54-61

A Survey of Analog-to-Digital Converters—Harry E. Burke Jr. (*Proc. I.R.E.*, vol. 41, pp. 1455-1462; Oct., 1953.) Basic principles of analog-to-digital conversion are described. The paper defines terms and introduces the subject in a clear and interesting manner. The treatment is elementary, starting with discussion of the terms "analog" and "digital" and clearly outlining the purpose of data conversion devices. An attempt is made to classify the many existing conversion devices into categories, and to point out the basic principles they have in common. Some representative techniques are presented as examples. Because of the survey nature of the paper, these examples are very brief and cover only the basic principles of the technique. The author does not include a discussion of accuracy, stability, linearity, or resolution for the devices covered, and no summary is given of the present state of the art. A sample data conversion system is illustrated to show the application of analog-to-digital conversion to automatic data reduction.

A. J. Winter

621.375.2 54-62

An Analog-to-Digital Converter for Serial Computing Machines — H. J. Gray, Jr., P. V. Levonian, and M. Rubinoff. (*Proc. I.R.E.*, vol. 41, pp. 1462-1465; Oct., 1953.) An analog-to-digital converter is described, the output of which can be read synchronously into a digital computer at better than a 1 megacycle rate. The analog voltage to be digitized is placed on the vertical deflection plates of a horizontally swept cathode ray tube. A quantized mask is inserted between a photocell and the cathode-ray tube. The horizontally swept electron beam generates a "cyclic" coded signal in the photoelectric circuits. The multiple ambiguities that could arise in the positioning of the beam using a binary coded mask are avoided by using a "cyclic" coded mask. Three alternate ways are shown of converting the "cyclic" to a binary code serially with the least significant digit first. The theorems underlying these conversions are proved.

Philip R. Westlake

ANALOG COMPONENT RESEARCH

54-63

Rotating Components for Automatic Control — Sidney Davis. (*Prod. Eng.*, vol. 24, pp. 129-160; Nov., 1953.) A survey of components commonly used in servomechanisms and analog computers is presented. The discussion is mainly of a qualitative and descriptive nature. However, figures are given on the accuracy and performance limits obtainable with most of the components. Some of the devices discussed are potentiometers both linear and nonlinear resistance variation and rotating and linear motion, nonlinear function generators, such as cams, non-circular gears, sine and cosine mechanisms, ball and disc integrators, synchros, resolvers, tachometers, servo motors, and miscellaneous elements. Several examples are given to show how these devices might be used in closed loop and open loop applications. The article is profusely illustrated and has excellent diagrams. (See also 54-64 of this issue.)

T. H. Bonn

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

ANALOG SYSTEMS RESEARCH

142 54-64
he Analog Computer — Dewitt H. ens. (*Prod. Eng.*, vol. 24, pp. 176- May, 1953.) The author presents a ussion of the logical design of og computers. There is a very brief ussion of Laplace Transform tech- e. The author then shows how with stors, capacitors, and operational lifiers the processes of addition, raction, integration, and differentia- can be performed. He shows connec- diagrams for an analog computer to e systems of simultaneous linear tions and also a linear differential tion with constant coefficients. The ection diagrams of computers to e several physical problems are vn. The application of analog tech- es to the study of the response of ed loop servomechanisms and the use uxiliary devices, such as recorders driving function generators, are also ussed. There is no mention of iplication or a means of integrating respect to a variable other than . This article forms an excellent panion article to "Rotating Compo- s for Automatic Control," Sidney is, *Prod. Eng.*, vol. 24, pp. 129-160; ember, 1953. (See 54-63 of this e.)

T. H. Bonn

142 54-65
The Relative Merits of A C and D C Signal Source in Analog Computers— H. Simpkin (*Elec. Eng.*, vol. 25, 230-233; June, 1953.) A comparison he advantages and disadvantages of and dc signals in a computer. Con- eration is given to the power supply uirements for each type signal. rc es of computing errors are de- bed for each system and the relative its of performing such functions as ition, multiplication, integration, and onometrical solutions are compared. olifier characteristics are also dis- sed, as are the response time and eral wiring problems. Combining both and dc signals in a computer is com- ated by the need for conversion ices and suitable power supplies for n types.

J. A. Fingerett

ANALOG EQUIPMENT

142 54-66
Electrical Analogues — G. Liebmann. t. *Jour. Appl. Phys.*, vol. 4, pp. 193- ; July, 1953.) A review of various ct electrical analogs for solving ndary-value problems. Author treats e types of analog device: (1) "Dis- uted parameter" (e.g., electrolytic ks); (2) "lumped parameter" (e.g., stance networks derived from the uivalent circuit" concept); (3) net- k analyzers of the ANACOM variety. iewer considers it an excellent ew, particularly useful for engineers not previously acquainted with these methods. Copious references to the literature are cited.

Courtesy of *Applied Mechanics Reviews*.

C. V. L. Smith

54-67
A Resistance-Network Analog Method for Solving Plane Stress Problems — G. Liebmann. (*Nature*, vol. 172, p. 78; July 11, 1953.) This letter to the editor briefly describes a resistance network method which solves the biharmonic equation. This is accomplished by cascading two networks which solve a partial differential equation. The mode of operation is described by the author as being the experimental counterpart of Southwell's relaxation method, but only at the boundaries since the networks are "self-relaxing" in their interior.

D. E. Hart

54-68
An Analog Reciprocal Function Unit For Use With Pulsed Signals — P. A. V. Thomas. (*Elec. Eng.* vol. 25, pp. 302-304; July, 1953.) A review of several methods of obtaining a reciprocal function is given. The function unit described is a monostable multivibrator which gives an output pulse width inversely proportional to the dc voltage applied to one grid. The circuit diagram is given with waveforms and experimental results.

J. A. Fingerett

54-69
Flight Simulators — K. H. Simpkin and E. T. Emms. (*Elec. Eng.*, vol. 25, pp. 270-273; July, 1953.) The flight simulator described reproduces the characteristics of a particular aircraft in contrast to previous simulators which were constructed with the viewpoint of generalized flying characteristics. Utilization of the cabin section of an actual aircraft increases the realism, and thereby, the training value. Severe emergency conditions can be simulated and flight crews may be given hours of practice in dealing with situations not justifiable in risk if attempted in the real aircraft. The flight computer is shown in block diagram.

J. A. Fingerett

681.142:519.272.1:534.44 54-70
The Correlatograph — W. R. Bennett. (*Bell Sys. Tech. Jour.*, vol. 32, pp. 1173-1185; September, 1953.) A paper-tape-recorder analog device for the continuous display of the short-term correlation function by means of a time/time-lag/correlation-factor plot (x-y-intensity) is described. It was designed for the analysis of 200-cps–4-kc signals in a magnetic-tape recording. The electromechanical design is based on the af spectrograph [3517 of 1946 (Koenig et al.)].

Courtesy of *Proc. I.R.E.* and *Wireless Engineer*.

UTILIZATION OF ANALOG EQUIPMENT

621.375.2 54-71
Application of Electronic Differential Analyzers to Engineering Problems — C. A. Meneley and C. D. Morrill. (*Proc. I.R.E.*, vol. 41, pp. 1487-1496; October, 1953.) This article is a useful introduction to, or review of, electronic differential analyzers of the dc analog type and their application to engineering problems. A section on the principal dc analog computing elements, such as integrators, summing amplifiers, and so forth includes a brief discussion of errors. A particularly interesting example of an application is the simulation of a servomechanism including its nonlinear characteristics such as static friction, potentiometer granularity, field saturation, etc; the performances of the simulated and actual system agree closely. The article ends with a list of other possible engineering applications and an estimate of the equipment required by a typical industrial installation.

G. A. Korn

621.375.2XR361.211 54-72
Analog Computing Applied to Noise Studies — R. R. Bennett. (*Proc. I.R.E.*, vol. 41, pp. 1509-1513; October, 1953.) An introduction into the theory of analysis of behavior of linear systems when driven by random stimuli has presented the task of determining the system weighting function (necessary for analytic investigation). This can be accomplished by using an analog computer in at least two different ways. First, by directly solving the defining equations or secondly, by solving the mathematically adjoint equations. The second approach yields the complete weighting function in one solution time, while the former requires many solutions, each of which provides a single value of the weighting function. The behavior of nonlinear systems, which do not obey the superposition theorem, must be investigated by a sampling process. Random forcing functions, coefficient values, and initial conditions are typical practical interesting situations. Methods for shaping the probability distributions and sampling the same are discussed. Formulas and graphs are presented to determine the number of runs required to give statistically significant results. Computer time, faster than real time, and analog-to-digital conversion of results for additional processing are used.

W. F. Gunning

DIGITAL COMPONENT RESEARCH

621.318.57:621.314.63 54-73
Semiconductor Diode Gates — L. W. Hussey. (*Bell Sys. Tech. Jour.*, vol. 32, pp. 1137-1154; September, 1953.) The general properties of gate circuits are discussed and a simple design analysis of transmission-type and switching-type gates is given. The results of an experimental check are noted, and the suitabili-

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

of point-contact and junction-type tifiers for different purposes is cussed.
urtesy of *Proc. I.R.E.* and *Wireless Engineer.*

82.12 54-74
The Phase-Bistable Transistor Cir-it — R. H. Baker, I. L. Lebow, R. H. diker, and I. S. Reed. (*Proc. I.R.E.*, . 41, pp. 1119-1124; September, 1953.) transistor circuit is described which forms the function of a bistable multi-brator, or flip-flop. However, the cir-it is phase-bistable rather than ampli-le bistable. The basic building block the transistor one-shot multivibrator. pointed out by the authors, the use a monostable transistor circuit, rather an a bistable circuit, overcomes the ect of the "hole-storage" phenomenon the switching time. Two one-shot ltivibrator circuits and six crystal te circuits compose one phase-bistable cuit. The construction of a binary unter and other logical circuits is scribed. Circuit details of the transis-circuits are given in full.

D. F. Rutland

54-75
Electronic Counter — E. Levey. *adio and Television News*, vol. 50, . 43-45, 114; September, 1953.) This icle describes construction details and eration of a simple four-stage binary unter and driver. It is a slow speed unter using 6SN7 dual-triodes. While is article is of no interest to the ex-rienced computer engineer, it would ovide an interesting experimental set-for the beginner or for an individual an allied field desiring to learn some-ing of computer operation.

D. E. Hart

54-76
Making Machines Remember — Ira M. ge. (*Prod. Eng.*, vol. 24, pp. 141-149; ril, 1953.) A survey of memory devices und in digital computers is presented. evices surveyed include acoustical lay lines, electromechanical devices uch as relays and embossed paper or astic), electronic devices (such as p-flops), electrostatic storage, capaci-storage, ferroelectric, ferroresonant, gnetic, mechanical, and optical orage devices. There are several seri-s errors in the article. For example, e switching time of flip-flops is given 20 micro-seconds. The switching time a magnetic core is given as 30 milli-conds. The signal retention of a mercu-delay line is characterized as poor ile the signal retention for barium anate crystals is characterized as od. Furthermore, there are many issions. For example, nothing is said the rectangular hysteresis properties ferroelectric materials. Some of the nclusions and statements of the con-lling considerations in choosing fferent types of storage are open to estion.

T. H. Bonn

621.375.2 54-77
An Analysis of Magnetic Shift Register Operation — Eugene A. Sands. (*Proc. I.R.E.*, vol. 41, pp. 993-999; August, 1953.) This paper gives sufficient information for designing a magnetic shift register using rectangular hysteresis loop material. Quantitative analysis of the register operation yields design criteria for such items as the minimum applied field strength required for switching, the turns ratio of input-output windings, the resistance in the transfer loop, and the maximum allowable leakage inductance. The analysis is based on equivalent circuits of the register and the author's previous analysis of the equivalent impedance of the cores (E. A. Sands, "The Behavior of Rectangular Hysteresis Loop Magnetic Materials under Current Pulse Conditions," *Proc. I.R.E.* vol. 40, pp. 1246-50; October, 1952). There is a discussion of the effects of leakage inductance on stability and speed of operation.

D. F. Rutland

54-78
Magnetic Drum Storage Devices — Robert L. Perkins. (*Prod. Eng.*, vol. 24, pp. 192-195; August, 1953.) Some of the magnetic drums which have been constructed by the Engineering Research Associates Division of Remington Rand, Inc. are discussed. The article concentrates on the specific techniques which have been used at Engineering Research Associates, including mechanical design of the drum and its housing and bearings, head design, the mechanism for spacing the head from the drum and the recording and playback signals. The article is brief and of a descriptive nature.

T. H. Bonn

54-79
Fused-Quartz Ultrasonic Delay-Line Memory — D. A. Spaeth, T. F. Rogers, and S. J. Johnson. (*Electronics*, vol. 26, pp. 151-153; December, 1953.) The research work performed on a fused-quartz type of solid ultrasonic delay-line memory is described. Various problems encountered in the design of solid delay-lines are discussed and reasons given for the selection of fused-quartz as the delay medium. Information pulses to be stored are converted to pulsed rf signals. The rf frequency is about 40 mc and the amplitude of signal into the delay-line is approximately 20 volts peak to peak. The pulsed rf from the delay-line is amplified by a wide band amplifier, detected and filtered and appears as a series of pulses at the output of the memory device. Spurious noise from the delay-line is shown to be 45 db down from the desired signal. Delay-lines of 300 and 400 microseconds length have been built and operated. A folded transmission path is used to obtain these delay times and still keep physical size small. Several references are given at the end of the article.

N. F. Loretz

54-80
A Diode-Capacitor Memory — (*Tech. Bull. Nat. Bur. Stand.*, vol. 37, pp. 171-173; November, 1953.) This memory, using essentially one capacitor and two semiconductor diodes per bit, can be designed for access to more than 100,000 randomly located words per second. Its speed, simplicity, designability, and probably high reliability are the factors to be weighed against its comparatively high cost of components. The basic principles are described; then the error-free performance of two experimental partial memories in a number of tests are cited. For further technical details the reader is referred to "An Experimental Rapid-Access Memory Using Diodes and Capacitors," by A. W. Holt, *Proc. Assoc. Comp. Mach.*, Toronto meeting, September, 1953.

Robert D. Elbourn

681.142:538.221 54-81
Digital Storage Using Ferromagnetic Materials — A. E. DeBarr. (*Elliott Jour.*, vol. 1, pp. 116-120; May, 1953.) Four memory devices for serial information are described: 1) *Magnetostriction Delay Line:* Nickel wire coiled into a space of 12 inches by 12 inches by ¼ inch provides a 1-millisecond delay for a train of 1.5-microsecond pulses spaced 1.5 microseconds apart. Acoustic waves, excited by magnetostriction, travel the length of the line and induce an output voltage in a pickup coil. Insertion loss is about 40 db. The line itself need not be ferromagnetic if a short nickel piece is brazed to the ends, or the ends are nickel plated. Temperature coefficients of delay have been obtained as low as 0.007 microsecond per millisecond per degree centrigrade. 2) *Static Magnetostriction Delay Line:* Same device as above with multiple pickup coils. The wire under each pickup coil is magnetized or demagnetized to represent binary ONE or ZERO. A single acoustic pulse travels down wire generating an output voltage in each coil. The coils, connected in series, deliver a serial train of pulses. A 25-word memory is described which has pulses spaced 3 microseconds. Reading is nondestructive. 3) *Magnetic Drum:* A 9-inch disc is used instead of a conventional drum to make head clearance less critical. Drum is forged brass, ferric-oxide coated. The heads are 5-mil Mumetal washers with a ¼-mil gap. Spaced 1 mil from side of disc rotating at 4500 rpm they develop 0.5 millivolt in 75 ohms. Digit packing of 170 per inch is used. 4) *Magnetic-Core Stepping Register:* A design similar to the original Harvard design is described. Using ferrites, a 10-core line has been operated at 300 kc. (See also similar article by A. E. DeBarr et al, in *Proc. Assoc. Comp. Mach.*, Jointly Sponsored by the ACM and the Mellon Institute, Pittsburgh, Pa., May 2-3, 1952.)

Dudley A. Buck

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

54-82
olycathode Counter Tube Applica- — J. H. L. McAsulan and K. J. ley. (*Electronics*, vol. 26, pp. 138- November, 1953.) This article dis- es the application of the Ericcson tron tube (polycathode gas dis- ge tube) to equipment which makes f decode scaler units. The operating iple of the polycathode counter tube a basic counter unit are described. matic diagrams of a millisecond , a batching counter and an oscil- pe time marker using the Dekatron he operation of these units are given. model of Dekatron tube is capable of ting at rates up to 20,000 per second. ral references are given at the end of rticle. Norman F. Loretz

54-83
Monoscope Tube for Computer and r Applications — John Hartmann. *c. Eng.*, vol. 73, pp. 208-212; h, 1954.) A monoscope type cathode tube designated K1043 is described echanical and electrical detail. Used onjunction with another CRT, this can be used to display characters screen. Thus it may be used as an ut device for digital computers. The oscope described here contains a le symbol on its target, so that n of e units are required for displaying n rent characters. This unit supplies a er output signal than monoscopes erly available. It may be used for r applications such as frequency iplying, pulse height analyzing, and e shifting. H. T. Larson

375.2 54-84
hotographic Techniques for Informa- Storage — G. W. King, G. W. Brown, L. N. Ridenour. (*Proc. I.R.E.*, vol. pp. 1421-1428; Oct., 1953.) This r comprises a discussion of the ntages of photographic techniques the storage of information. Chief ng these advantages are the high lable storage density and ease of -out. Of especial interest are the ral ingenious schemes of optical -out proposed by the authors. These well worthy of close attention and doubtless applicable in other con- . The chief disadvantage of ographic storage techniques is that information may not be written over ld information, and that the informa- cannot be modified bit-by-bit but r must be renewed block-by-block. disadvantage is recognized by the rs who nevertheless conclude that technique is useful in situations e large amounts of slowly varying mation are to be processed. F. A. Schwertz

DIGITAL SYSTEMS RESEARCH

54-85
erial Digital Adders for a Variable ix of Notation — R. Townsend. *c. Eng.*, vol. 25, pp. 410-416; ber, 1953.) A description is given of different methods of adding numbers in a digital computer, which are represented in serial coded groups, the digit within each group being in serial binary code. Each group can portray a number with a different radix of notation, so that quantities having a variable radix such as ounces, feet, tons, and so forth can be represented. The radix of notation of the groups may be changed at will during the course of the program. R. G. Canning

681.142 54-86
Micro-Programming and the Design of the Control Circuits in an Electronic Digital Computer — M. V. Wilkes and J. B. Stringer. (*Proc. Camb. Phil. Soc.*, vol. 49, part 2, pp. 230-238; April, 1953.) The authors propose an approach to the design of the control unit of an electronic digital computer, essentially by regarding the unit as a complete miniature machine controlled by a "micro-program." The latter may be built up from "micro-subroutines" and its task is to cause the control unit to obey the ordinary program in the large machine. The micro-program differs from ordinary programs in that it is permanent and can therefore be held in a static (unreceptive) store. The "micro-orders" of which the micro-program is composed are in a code chosen mainly for speedy and simple execution by the "micro-control unit". Thus each micro-order specifies the micro-address of that which is to follow; the remaining binary digits of the micro-order are associated individually with gates in the arithmetical and control units. A possible system for a parallel machine is described in general terms; the micro-program is supposed to be wired on a diode matrix. A micro-program of 38 micro-orders is given as an example. The purpose of the proposal is to permit the provision of elaborate built-in facilities (e.g. floating-point) with logically simple circuits, the complexity residing in the micro-program. This it does, but the authors' claim that this method of designing the control circuits is "wholly logical" and "enables alterations or additions to the order code to be made without *ad hoc* alterations to the circuits" seems a little too sweeping. The usefulness of the idea depends on having a small static store with access time of the same order as transfer times within the arithmetical unit. Since this paper was written, magnetic cores have come to rival the diode matrix for this purpose. S. Gill

621.385X621.375.2 54-87
An Automatic Telephone System Employing Magnetic Drum Memory — W. A. Malthaner and H. E. Vaughan. (*Proc. I.R.E.*, vol. 41, pp. 1341-1347; October, 1953.) After tracing the resemblance between the functions of a computer and those of a telephone exchange common control the authors describe an experimental system which goes part of the way towards the realization of the telephone system envisaged by Dr. Lewis. In their system the conditions of every subscriber's line are continuously scanned and changes in the line conditions are recorded on magnetic drums. Dialled information is assembled and passed to other drums which allocate a route suitable for the call and retain a memory of the completed connexion. The speaking paths through the exchange are, in effect, slave mechanisms and their release is also controlled from the common drums. The system is a development of the well known common control system made possible by the exploitation of the high speed technique and large capacity storage devices of electronic computers which enable fewer controls to be used then hitherto. It is interesting to note that even when relatively slow devices such as magnetic drums are used, the designers have to use sequence circuits having a response time of fractions of a microsecond. Full exploitation of common control techniques can only come when even higher speeds can be obtained economically and reliably, but this is probably only a question of time. S. W. Broadhurst

621.375.2 54-88
Electronic Computers and Telephone Switching — W. D. Lewis, (*Proc. I.R.E.*, vol. 41, pp. 1242-1244; October, 1953.) From its inception automatic telephony has posed problems analagous to those of digital computing but as the author points out, the solutions to the problems have been found with the aid of slow acting devices. These devices have however been characterized by an extreme reliability not yet obtainable with computers. Nevertheless, the logical development of the principle of common control, already well known in the switching art, may lead to the use of computer techniques to develop the equivalent of a robot operator capable of handling all the traffic in a telephone exchange. S. W. Broadhurst

DIGITAL EQUIPMENT

54-89
An Electronic Batching Counter — R. T. Craxton. (*Elec. Eng.*, vol. 25, pp. 424-426; October, 1953.) The article describes a pre-determined electronic digital counter for industrial applications. The unit uses Dekatron counting tubes, cold-cathode gas-filled tubes, and achieves a maximum batching speed of 40,000 counts per minute. R. G. Canning

54-90
FOSDIC, a Film Optical Sensing Device for Input to Computers — (*Tech. Bull. Nat. Bur. Stand.*, vol. 38, pp. 24-27; February, 1954.) The input to FOSDIC is a microfilm copy of Census enumerator's sheets marked with an ordinary pen or pencil. Its output is a magnetic tape suitable for reading into a UNIVAC computer. It uses a cathode-ray tube as the source of a spot of light which scans the film, and a photocell which senses the marks. Specially printed bars on the original document enable FOSDIC to locate the columns it should scan for marks, thus it can tolerate considerable mis-alignment of the document on the microfilm. A frame of film may contain

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

nage of a sheet 14 by 16 inches ining 2800 marks and can be ed in 1¼ seconds; thus the informa- ate is about 2000 binary digits per d. The equipment occupies four h-high relay rack-sized cabinets.
Robert D. Elbourn

75.2XR257 **54-91**
chine Aid for Switching Circuit n — Claude E. Shannon and Edward oore. (*Proc. I.R.E.*, vol. 41, pp. 1351; October, 1953.) This paper ibes a machine which is an aid in esign of switching circuits. The ular machine which is described relays, selector switches, gas s, and germanium diodes and works circuits of four variables. To use achine a proposed circuit is set up plugboard and $2^4 = 16$ switches are ccording to the specifications re- of the circuit. The machine can verify the correctness of the circuit heck to see if any of the contacts al elements) can be eliminated by permanently "shorting" or nently "opening" the contacts. it can obtain a mathematically us lower bound for the number ype of contacts needed to satisfy pecifications. The detailed circuit n of the machine is not discussed. advantages of a special purpose ne to consider problems of logical ts as compared to general purpose nes are discussed. R. E. Meagher

54-92
e Gottingen Electronic Calculating nes I, II (in German) — L. Bierman, lling, W. Hopmann, and A. Schluter. *M* 33 1/2, pp. 48-60; January- ary, 1953.) The authors describe a al-purpose electronic automatically nced digital computer and compare other mathematical machines. The uter is controlled by a perforated tape and has a magnetic drum ry which rotates at 50 rps. It is a computer containing less than 500 m tubes and 100 relays. The com- has provision for cyclic inter- e of numbers stored on the magnetic Paper concludes with an example, is the program for the numerical ation of a differential equation. esy *Applied Mechanics Reviews.*
C. L. Perry

54-93
e OARAC — B. H. Geyer. (*Radio Television News*, vol. 50, pp. 83, 81; October, 1953.) This is a semi- ical article which describes fically — the characteristics of the AC, a general purpose magnetic drum uter, and generally — the technique ogramming a simple problem for a l computer with an example using OARAC code list. Good background ng for the beginner. D. E. Hart

54-94
g Brain to Solve Aviation Problems— p Klass. (*Aviation Week*, vol. 58, 51-66; February 9, 1953.) The Air Force's large scale digital computer, OARAC, built by General Electric and installed at Wright Air Development Center, Dayton, Ohio, is described in a semi-technical article by Philip Klass. Except for a number of minor errors, the article gives a good picture of the design and construction characteristics of the machine. According to the article, the machine is outstanding for its 10,000 ten decimal digit word memory, for its simplicity of design, and for its ease of maintenance. Input and output are by means of magnetic tapes, but complete control is possible from an extensive front panel. A number of check circuits and an automatic "roll-back" are built into the machine.
E. P. Little

54-95
SEAC, Improvements Increase Computing Power — '*Tech. Bull. Nat. Bur. Stand.*, vol. 38, pp. 8-13; Jan., 1954.) Additions to SEAC have included a 512-word, cathode-ray tube memory, a number of magnetic wire input-output units and magnetic tape auxiliary storage units, five additional mathematical operations, an alternative three-address instruction system with relative address features, and facilities for automatically monitoring a program by printing each instruction and its result, or only "negative" instructions and their results. Half of the type 6AN5 tubes still meet rather stringent tests after 8,700 hours service. Only 0.02 per cent of the germanium diodes cause machine malfunction per 1000 hours. In a typical month SEAC works on as many as 50 different problems. Operating efficiency averages 72 per cent. Improvement in cathode-ray tube memory has been achieved by developing better techniques for detecting and removing flaws and by promoting the manufacture of better tubes. (See 54-38 of March, 1954.)
Robert D. Elbourn

54-96
SWAC, National Bureau of Standards Western Automatic Computer — (*Tech. Bull. Nat. Bur. Stand.*, vol. 37, pp. 145-150; October, 1953.) SWAC is a parallel type computer with a 256-word cathode-ray tube memory and a 4096-word magnetic drum auxiliary memory. It can perform 16,000 additions or 2,600 multiplications per second. Waiting time for information from the drum is saved by storing serially and transferring all 32 words in a channel in a single revolution starting with any word. Paper tape and punched cards are used for input and output. SWAC contains 2,600 tubes and 3,700 crystel diodes. The average tube life is between 8000 and 10,000 hours. Most failures are from low emission or intermittent shorts. A new converting output instruction converts fractional binary numbers to octonary, decimal, or any other base up to 16. Problems studied include systems of linear equations, associated legendre functions, "Monte Carlo" methods, biological survival probabilities, atmospheric circulation, combinatorial analysis, Fourier synthesis of X-ray diffraction patterns, and the primality of Mersenne numbers.
Robert D. Elbourn

621.375.2 **54-97**
Electronic Circuits of the NAREC Computer — Paul C. Sherertz. (*Proc. I.R.E.*, vol. 41, pp. 1313-1320; October, 1953.) The NAREC, Naval Research Laboratory computer, is a parallel computer and in logical structure it could be classified as belonging to the IAS (The Institute for Advanced Study) family. However, certain engineering design principles, fundamental circuitry and physical layout of the computer have much variance with other computers belonging to the IAS family, namely: 1) Crystal diodes are used in place of vacuum tubes whenever a substantial savings of space or power could be anticipated. 2) A basic asymmetrical low output impedance flip-flop circuit is used in the arithmetic unit and control. Counters, steppers, inverters and adders are built with the same basic circuit principle. 3) Plug in units are used extensively throughout the computer. Its engineering design principle which is common to the IAS family of computers is: "Where practical, circuits should be designed for DC operation and aperiodic triggering." As the title implies, this paper deals mainly with the electronic circuits of the arithmetic unit and control. Detail circuits, timing charts and logical diagrams are given. Its electrostatic memory system, magnetic drum, order list, etc., are not described.
J. D. Chu

321.375.2 **54-98**
Engineering Description of the IBM Type 701 Computer — Clarence E. Frizzell. (*Proc. I.R.E.*, vol. 41, pp. 1275-1287; Oct., 1953.) The 701 uses an electrostatic memory with magnetic tapes and drums for auxiliary storage. Internal operation is parallel-synchronous, with one megacycle pulse rate and 12 microsecond operation cycle. Operating time is materially reduced by parallelling memory regeneration with multiplication and other operations not involving the memory. Magnetic tape speed is 75 inches per second, using six data channels and a check channel. Drums are forged aluminum cylinders wound with Cunife wire and ground to a smooth surface. Non-return-to-zero recording is used on tapes and discrete pulse recording on drums. The latter use a counter-type address location system. Other auxiliary equipment includes a card reader, a high speed punch, and a line printer which is self-checking through signals fed back from type wheels. A simplified control panel provides visual display of data in the various registers and controls for manual insertion of data and step-by-step operation. Facilities are available for marginal checking by varying pulse rate and power supply voltages. Unitized construction permits several test crews

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

rk simultaneously on different units. e circuit design was started, engi- were provided with complete oper- manuals explaining machine logic. led installation and maintenance als have also been prepared.

R. F. Shaw

UTILIZATION OF DIGITAL EQUIPMENT

54-99

ilization of Digital Computing ines — (*Nature*, vol. 172, pp. 649- Oct. 10, 1953.) This is a summary of talks presented at a meeting of the sh Association by F. C. Williams R. K. Livesley, University of hester, and G. G. Alway, National ical Laboratory. Included are dis- ons of the advantages of automatic utation, types of problems which been programmed, and specific ems.

D. E. Hart

54-100

utomatic Computing in Aircraft Engi- ng — G. G. Alway. (*Engineering*, 176, p. 351; September 1953.) This presents a short discussion of the cation of the digital machine at the nal Physical Laboratory, the pilot E., to engineering design problems. computer has been most successful aling with problems concerned with x algebra. Its capabilities in this are given. Two programs for the ion of flutter problems on the pilot E. have been made so far. A brief ription of these programs is pre- d. Emphasis is placed on two ns for automatic digital computers ning more powerful the more they sed. First, a library of subroutines ilt up which facilitates the program- of new problems; and second, ammers continually improve the iques for using their machine. esy of *Applied Mechanics Reviews*.

L. D. Findley

54-101

omputing Machines Input and Out- — R. Bird. (*Elec. Eng.*, vol. 25, 407-410; Oct. 1953.) The use of uters in business applications asizes the need for high speed in- nd output. Digital computers with a binary scale of representation nally are assumed, while their and output requirements are usually nal and sometimes variable radix ion. Methods are described for the ersion of conventional punched card into the binary scale and the re- ersion of the computer output from y to a form suitable for the operation printer or card punch.

R. G. Canning

54-102

vings and Mortgage Division, Ameri- Bankers Association: Report of the nittee on Electronics, September (slightly shortened) — Joseph E. Perry, Gustave Bottner, Walter F. Clow, Robert F. Marchant. (*Computers and Automation*, vol. 3, pp. 10-12; January, 1954) This is the first report of a committee set up to serve as an information link between the manufacturers and the users of automatic machines performing banking operations. The report describes some of the general characteristics of the banking operation that specifically relate to methods of mechanizing the individual functions performed, at the same time providing adequate proofs and audit control. Assuming that equipment can be produced to carry out any desired function, a survey of bankers' requirements was collected and is summarized. This indicates the desire for a wide variety of equipment for central operations, for operations performed at individual bank stations, and for the communication of these central and remote equipments to provide an integrated system with centralized control over all operations and rapid distribution of pertinent information to the individual stations.

G. E. Gourrich

54-103

Language Translation by Machine — A Report of the First Successful Trial — Neil Macdonald. (*Computers and Automation*, vol. 3, pp. 6-10; February, 1954) In January, 1954 "the first successful demonstration of meaningful translation from one language to another language by machine" was carried out on the IBM 701. This was the result of the solution of a rather restricted translation problem, having a vocabulary of 250 Russian words with English equivalents, which was carried out jointly by the Institute of Languages and Linguistics of Georgetown University, Washington, D.C., and the IBM Corporation. The author describes the nature of the trial, the form of the dictionary, and the six rules of operational syntax upon which the translation is based, together with a flow diagram of the operational translation procedure. A brief description of how the translation is carried out is also included. The historical background of the project is presented, leading to a discussion of the possible future import of machine translation. No discussion is given of the relation of the described project to other work of a similar nature.

G. E. Gourrich

621.375.2 54-104

Frequency Analysis of Digital Computers Operating in Real Time — J. M. Salzer. (*Proc. I.R.E.*, vol. 42, pp. 457-466; February, 1954.) This paper deals with the digital computer as used in control systems. It is assumed that the data input to the computer are sampled at a constant rate, the time delay in computation is negligible, the computer relates a single output quantity to a single input quantity and that the computer performs only certain "linear operations", in real time. From these assumptions, the analysis reveals the resulting amplitude and phase characteristics that the computer may exhibit with various programs and the conditions for stability and realization of the programs. Throughout the paper the digital computer with its program is analyzed as a filter along lines familiar in network theory and servo design.

D. F. Rutland

ORIENTATION READING

54-105

The End of an Epoch: The Joint Computer Conference, Washington, D.C., December 1953 — Alston S. Householder. (*Computers and Automation*, vol. 3, pp. 6-7; January, 1954) In the author's opinion the conference in Washington, D.C., with its emphasis on reliability and operation of existing computing equipment, indicates the end of the successful era of realizing the "great idea" of the large high-speed digital computer. On the one hand, the computer field is disintegrating into many smaller very specialized areas, while on the other hand many people are turning to the anticipation of the next "great idea" in fields such as business applications and automation. The author feels that these new ideas have not yet been realized, and effort should be spent on producing such advances instead of working so extensively with the developments now at hand. A summary of some of the topics discussed at the conference is included.

G. E. Gourrich

621.375.2 54-106

Computing Bit by Bit, or Digital Computers Made Easy—Arthur L. Samuel. (*Proc. I.R.E.*, vol. 41, pp. 1223-1230; October, 1953.) This paper, written in simple, nontechnical language, establishes the need of science and industry for automatic calculating devices and proceeds to explain the "physiology" or functioning of such machines. This explanation provides a basis for determining the future role of the computer and lays the groundwork for an understanding of the more highly technical papers which follow. With particular emphasis on problems of computer storage (or memory) and allied operational problems, the paper surveys briefly but comprehensively those fields of computer research and development which appear to offer greatest promise for further investigation and improvement. The nonspecialist reader must certainly agree with the author that "Computers are here to stay, and it is high time for us to be learning more about them."

John B. Bennett

54-107

Glossary of Terms in the field of Computers and Automation—Discussion— I., Alston S. Householder; II. E.C. Berkeley. (*Computers and Automation*, vol. 3, pp. 22-24; February, 1954) Referring to the glossary of terms published in past issues of *Computers and Automation*, Mr. Householder suggests several ground

This page has been left blank in order that readers may mount all reviews on cards.
— *The Editor*

s for the editor to follow in revising supplementing the glossary in the re, since a glossary may be more ul with careful editing than if set up ely to report all current usage. Comts are made on several terms of the ssary and several new terms are gested, in some cases resolving iguous usage, and in others reporting terminology of the Oak Ridge Labora-. Mr. Berkeley suggests methods for editor of the glossary to be more in h with current usage and standards erminology.

G. E. Gourrich

54-108

Reflective Thinking in Machines — L. Gruenberg. (*Computers and Autoion*, vol. 3, pp. 12-19,26,28; Februa- 1954) "Reflective thinking," in the se of John Dewey's interpretation, is thought process useful to the solu- of man's problems and therefore uld be investigated for possible hanization. An analysis of the basic nents required to handle information he thought process shows that comng machines possessing memory and means for standardizing meaning sess the raw capacities with which perform reflective thinking. Furthere, an analysis of the functions carried during the five stages of reflective king (situation recognition, problem ling, search for solutions, determining t solution, and test by action) can be olved into operations which can be formed by electronic or mechanical ices of contemporary design. In fact, chines already have been built which ry out some of these functions. Unafted by man's basic drives and emons, machines would think more deadably and precisely. Machines must built to carry out certain of the funcns of reflective thought in order to be with continuous increase of man's blems. How much control the operator st have over a machine depends on its abilities for following the process of lective thought in its entirety. How ch control should be maintained by operator depends on the consences of autonomous operation. There many unanswered questions as to w such machines would behave, the swers to which may provide insight o human thought. This paper has more phasis on the analysis of the thought cess than in methods of mechanization. veral references are included.

G. E. Gourrich

54-109

The Facts About Automation — H. F. ver. (*Prod. Eng.*, vol. 24, pp. 129-133; cember, 1953.) This article is in the m of a question and answer session ween *Product Engineering* and Mr. ver. Mr. Dever points out that industry now spending two to three times the percentage of its income that it did ten years ago on instruments, and that the percentage is still increasing. He points out that economy, safety and quality are the factors which lead toward automation. He further states that a major problem in effective development of automation will be systems engineering, namely, integrating design of the product, instrumentation and production machinery, and servo mechanisms into an automatic factory. According to the author, computers will be very useful but they should be smaller and more specialized than at present, and have improved input and output devices.

T. H. Bonn

54-110

Automation in the Kitchen — Fletcher Pratt. (*Computers and Automation*, vol. 3, pp. 13, 26; January, 1954) Use of a punched paper tape "recipe" to control automatic or semiautomatic food preparation devices, such as electric ranges, mixers, etc., would facilitate precise and reproducible production of cooked food by inexperienced operators in both private and commercial kitchens. Deep freezing and subsequent shipment at some intermediate point in the process would allow the individual housewife to easily prepare a large variety of dishes originating all over the world. No system is proposed for the mechanization of these ideas.

G. E. Gourrich

BOOK REVIEWS

54-111

Faster Than Thought — edited by B. V. Bowden. (Sir Isaac Pitman and Sons, Ltd., London, England, 416 pp.; 1953.) A British work on the opportunities which electronic digital computers offer and the difficulties associated with them, including material on computers in America. History and theory, organization, construction, and programming are considered, as is the value of the digital computer to the design engineer, and applications to business, commerce, government, and various types of research.

Courtesy of *Electrical Engineering*.

(For a more detailed review written by W. Buchholz see *Proc. I.R.E.*, vol. 41, pp. 1550-1551; October, 1953.—Editor.)

54-112

Computing Mechanisms and Linkages— Antonin Svoboda. (*M.I.T. Rad. Lab. Ser.*, McGraw-Hill Book Co., New York, N.Y., 359 pp., 177 figs.; 1948.) This book is devoted to computers made entirely of mechanical elements. Such computers have been used for the rapid solution of gunfire control problems, in navigation and sometimes in other engineering applications. They are still the most economical, reliable and sturdy machines in those fields in which an accuracy of one part in a thousand is sufficient. A historical introduction describes the adding, multiplying, integrating, resolving and transforming components which have been customarily used in mechanical computers employing gears, cams, friction drives, worm drives, and linkage mechanisms. But the principal concern of this book is a detailed study of the simplest of the mechanical methods, namely, the linkage mechanisms made entirely of rigid bars joined by pivots. The design of linkage mechanisms has been an empirical art in which the principal techniques were graphic. The largest part of this book is devoted to an exposition of these graphic techniques of which very few have been published previously. Good design of linkage computers requires a proper compromise between the complexity of the mechanism to make it follow the variations of an analytical or an empirical function and the need for simplicity to reduce the accumulation of errors due to play in the hinges. The mathematical part of the problem is usually the determination of the parameters to allow the solution to pass through a sufficiently scattered set of arbitrary points on the desired curve with the hope that the derived curve does not depart too much at other points. A harmonic transformer consists of a crank driving a sliding rod. By varying the lengths of the crank and rod, the pivot points and the initial points of the motions, one obtains an enormous selection of available transformations. Convenient tables are given in a 50-page appendix for the selection of the desired function. Combinations of two or more harmonic transformers give closer approximations. A study of the effect of structural errors on the final accuracy of the mechanism is included. The three-bar linkage (more frequently called the four-bar linkage) is then investigated. This permits more types of curves and functions since there are more parameters. Numerical tables and graphic diagrams of families of curves and nomograms are given to enable first-order design. Combinations of two and three harmonic transformers and a three-bar linkage are considered in several numerical examples. Functions of two independent variables are mechanized approximately by means of linkages having two degrees of freedom. These are simpler and less expensive than the three-dimensional cams frequently used for this purpose. Since the book is essentially a manual for designers, very few references are made to the literature. Important early work in the mathematical theory of approximating linkages was done by Burmester and Chebyshev. There has been much recent work by Z. S. Bloh and N. G. Bruyevich.

Courtesy of *Applied Mechanics Reviews*.

M. Goldberg

INSTITUTIONAL LISTINGS

IRE Professional Group on Electronic Computers appreciates the support given by the organizations listed below. listing of products and services is limited in length and therefore is not necessarily all-inclusive.

BELL TELEPHONE LABORATORIES, INC.
463 West St., New York 14, N. Y.

Research and Development for the
Bell System and the Armed Forces

BURROUGHS CORP., Electronic Instruments Division
1209 Vine Street, Philadelphia 7, Pa.

ulse Control Equipment and Systems; Special-Purpose Tubes;
and Computation Services

COMPUTING DEVICES OF CANADA LIMITED
P. O. Box 508, Ottawa, Ontario, Canada

Digital & Analog Computers, Automatic Control Devices,
Servomechanisms, Research

ELECTRONIC ASSOCIATES, INC.
Long Branch, New Jersey

Analog Computers, Computer Components, Vari-plotter
Plotting Boards, DC Resolvers

FAIRCHILD CAMERA AND INSTRUMENT CORP.
POTENTIOMETER DIVISION
225 Park Ave., Hicksville, L. I., N. Y.

Linear and Non-Linear Precision Potentiometers
Single, Multi-Turn, Wirewound and FilmPots

HUGHES RESEARCH AND DEVELOPMENT LABORATORIES
Culver City, California

Electronic Equipment — Radar,
Guided Missiles, Computers

INTERNATIONAL BUSINESS MACHINES CORP.
590 Madison Ave., New York 22, N. Y.

Electronic Computers,
Scientific Computing Service

LIBRASCOPE INCORPORATED
1607 Flower Street, Glendale 1, California

Analog and Digital Computers for
Industrial Application; Analog Components

(Please see outside back cover for additional listings.)

INSTITUTIONAL LISTINGS (Continued)

MAGNETICS RESEARCH COMPANY
142 King Street, Chappaqua, New York

Magnetic Shift Registers, Magnetic Switching Systems, Pulse Transformers

THE W. L. MAXSON CORPORATION
460 West 34th St., New York 1, N. Y.

Research, Development and Production Computers, Radars, Servos, & Systems

MID-CENTURY INSTRUMATIC CORP.
611 Broadway, New York 12, N. Y.

Analog Computers - Electronic Function Generators - Recorders - Servomechanisms

POLYTECHNIC RESEARCH & DEVELOPMENT CO., INC.
55 Johnson Street, Brooklyn 1, New York

Magnetic Amplifiers for Servo Systems, Voltage Regulators, Speed Regulators

RAYTHEON MANUFACTURING COMPANY
Waltham 54, Massachusetts

Data Handling & Computing Systems, Magnetic Components, Tape Drives, Computing Services

REEVES INSTRUMENT CORPORATION
215 East Ninety-first St., New York 28, N. Y.

Electronic Analog Computers - Servomechanism Components - Resolvers - Gyros - Gears

REMINGTON RAND, INC.
315 Fourth Avenue, New York 10, N. Y.

Univac and E. R. A. Systems . . . Computing Centers . . . and Training Courses

TECHNITROL ENGINEERING COMPANY
2751 No. Fourth St., Philadelphia 33, Pa.

Digital Computers, Memories, Pulse Transformers, Delay Lines

THE TELEREGISTER CORPORATION
443 Fairfield Avenue, Stamford, Connecticut

Development — Data Handling and Inventory Systems — Digital and Analog Computers

The charge for an Institutional Listing is $20.00 per issue or $60.00 for four consecutive issues. (The number of characters, including spaces, is limited to ninety for the company name and address, and ninety for the products and services.)

Applications for Institutional Listings and checks (made out to the Institute of Radio Engineers) should be sent to Technical Secretary, Institute of Radio Engineers, 1 East 79th St., New York 21, N. Y.